

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних, розрахунково-графічних
та самостійних робіт з дисципліни

ОСНОВИ МОДЕЛЮВАННЯ
СКЛАДНИХ СИСТЕМ

*(для студентів 2 курсу денної та заочної форм навчання
напряму підготовки 6.080101 – Геодезія, картографія та землеустрій)*

Методичні вказівки до виконання лабораторних, розрахунково-графічних та самостійних робіт з дисципліни «Основи моделювання складних систем» (для студентів 2 курсу денної та заочної форм навчання напряму підготовки 6.080101 – Геодезія, картографія та землеустрій) / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова; уклад.: І. С. Творошенко. – Харків : ХНУМГ ім. О. М. Бекетова, 2015. – 99 с.

Укладач І. С. Творошенко

Рецензент д.е.н., проф. К. А. Мамонов

Рекомендовано кафедрою геоінформаційних систем, оцінки землі та нерухомого майна, протокол № 1 від 29 серпня 2014 р.

ЗМІСТ

ВСТУП.....	4
1 ВКАЗІВКИ ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ.....	5
1.1 Використання інструментальних засобів для опису функціональних моделей об'єктів та процесів.....	5
1.1.1 Створення функціональної моделі за допомогою ERwin.....	6
1.1.2 Декомпозиція контекстної діаграми.....	24
1.1.3 Діаграма декомпозиції другого рівня.....	28
1.2 Використання інструментальних засобів для моделювання потоків даних.....	33
1.3 Використання інструментальних засобів для опису логіки взаємодії компонентів об'єктів та процесів.....	45
1.4 Використання інструментальних засобів для опису змішаної моделі об'єктів та процесів.....	63
1.5 Ознайомлення з інструментальними засобами для опису процедур автоматизованої побудови мережевих моделей ГІС.....	70
2 ВКАЗІВКИ ДО ВИКОНАННЯ РОЗРАХУНКОВО-ГРАФІЧНИХ РОБІТ..	78
2.1 Структура розрахунково-графічних робіт.....	78
2.2 Правила оформлення розрахунково-графічних робіт.....	79
2.3 Приклади бібліографічного опису в переліку посилань	82
2.4 Критерії оцінювання розрахунково-графічних робіт	83
3 ВКАЗІВКИ ДО ВИКОНАННЯ САМОСТІЙНИХ РОБІТ.....	84
3.1 Основні поняття нечіткості.....	84
3.2 Процес і система нечіткого логічного виведення.....	88
3.3 Методи нечіткого виведення.....	90
3.4 Метод резолюцій.....	94
3.5 Експертні системи.....	94
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	98

ВСТУП

У даний час не можна назвати галузь людської діяльності, в якій у тій чи іншій мірі не використовувалися б методи моделювання. Особливо це стосується сфери управління різними системами, де основними є процеси прийняття рішень на основі одержаної інформації.

Гіпотези та аналогії, що відображають реальний, об'єктивно існуючий світ, повинні володіти наочністю або зводитися до зручних для дослідження логічних схем, що спрощують міркування, логічні побудови та дозволяють проводити експерименти, уточнюючи природу явищ, називаються *моделями*.

Модель – це об'єкт-заступник об'єкта-оригіналу, що забезпечує вивчення деяких властивостей оригіналу. Заміщення одного об'єкта іншим для одержання інформації про найважливіші властивості об'єкта-оригіналу за допомогою об'єкта-моделі називається *моделюванням*.

Таким чином, моделювання може бути визначене як уявлення об'єкта моделлю для отримання інформації про цей об'єкт шляхом проведення експериментів з його моделлю. Теорія заміщення одних об'єктів (оригіналів) іншими об'єктами (моделями) і дослідження властивостей об'єктів на їхніх моделях називається *теорією моделювання*.

Дисципліну «Основи моделювання складних систем» студенти вивчають на 2 курсі денної і заочної форм навчання напряму підготовки 6.080101 «Геодезія, картографія та землеустрій».

Методичні вказівки охоплюють основні розділи дисципліни з моделювання складних систем та мають на меті сприяти ознайомленню студентів з інструментальними засобами візуального моделювання.

Методичні вказівки написані з таким розрахунком, щоб студенти вивчили теоретичний матеріал за рекомендованою літературою, виконали лабораторні роботи за методологіями нотацій IDEF0, DFD, IDEF3 у процесі моделювання для заданої предметної області за допомогою сучасного інструментального середовища, змогли самостійно зробити розрахунково-графічні роботи, опрацювали теми самостійної роботи та підготувалися до екзамену з даної дисципліни.

1 ВКАЗІВКИ ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

1.1 Використання інструментальних засобів для опису функціональних моделей об'єктів та процесів

Мета роботи: ознайомитися з пакетом візуального моделювання ERwin. Освоїти його застосування для побудови функціональних моделей.

Методичні вказівки по організації роботи студентів

Інструментальне середовище ERwin.

Головне вікно, що відкривається за замовчуванням при запуску ERwin (рис. 1), містить такі елементи палітри інструментів (вибір панелі інструментів залежить від обраної нотації).

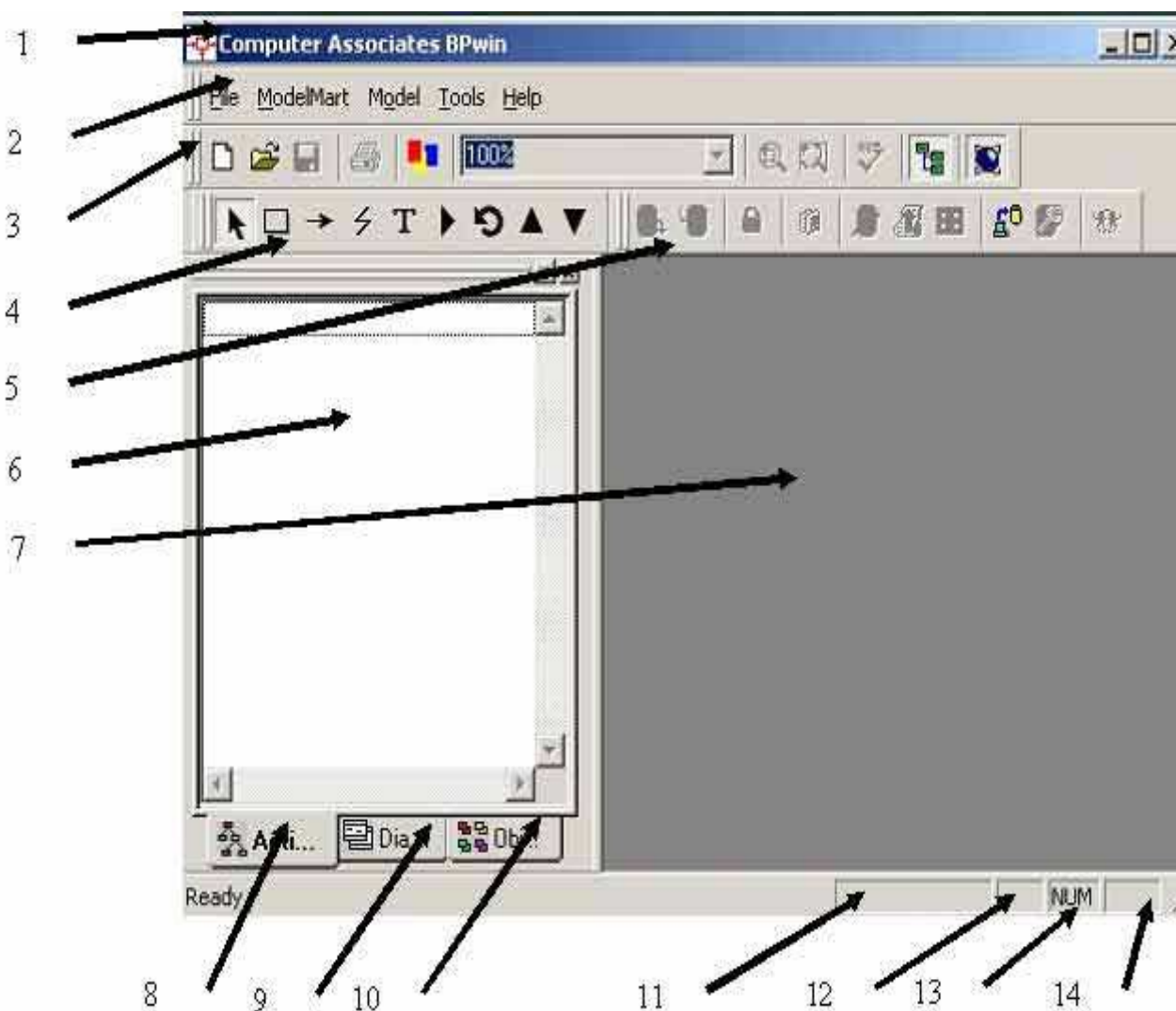


Рисунок 1 – Головне вікно ERwin

1 – інформаційний рядок, що містить назву продукту та кнопки мінімізації, відновлення і закриття додатку.

2 – рядок меню. Компоненти цього рядка відповідають додатку Windows і забезпечують доступ до всіх функцій ERwin.

3 – панель інструментів «Стандартна». На панелі інструментів «Стандартна» знаходяться кнопки, що забезпечують швидкий запуск часто виконуваних задач. При вказівці на кнопку, поруч з нею з'являється підказка з найменуванням кнопки.

4 – панель інструментів ERwin залежно від обраної методології створення діаграм в головному вікні ERwin можуть бути показані інструменти:

- для моделей згідно з методологією IDEF0;
- для моделей згідно з методологією DFD;
- для моделей за методології IDEF3.

5 – панель інструментів **Model Mart**. Кнопки на панелі інструментів ModelMart застосовуються для організації групової роботи й зв'язаних задач.

6 – навігатор моделі **Model Explorer**.

7 – графічне поле. Якщо на екрані відображається навігатор, то графічне поле розташоване в правій частині головного вікна ERwin. При відсутності браузерa графічне поле є повною областю вікна. У цій області створюються й редагуються діаграми ERwin. Подвійне клацання по графічній області відкриває вікно властивостей для поточної діаграми.

8 – кнопка Model Explorer у режимі Activity.

9 – кнопка Model Explorer у режимі Diagram Tree.

10 – кнопка Model Explorer у режимі Object Tree.

11, 12, 13, 14 – статусний рядок, подає інформацію про основні опції меню та кнопки панелей інструментів.

1.1.1 Створення функціональної моделі за допомогою ERwin


Методика створення функціональної моделі.

Розглядається модель процесу «Побудова будинку»:

- підготовка до будівництва;
- будівництво стін (каркасу будинку);
- проведення внутрішніх та зовнішніх робіт;
- здача будинку.

Будівельна компанія, маючи дозвіл на будівництво, матеріал та кошти, має на маті отримати збудований дім.

Запустити **ERwin** (Кнопка Start / ERwin). Якщо з'являється діалог **ModelMart Connection Manager**, натисніть на кнопку **Cancel** (Скасування).

Клацніть по кнопці .

З'являється діалогове вікно **I would like to** (рис. 2). Внесіть у текстове поле **Name** ім'я моделі процесу «Побудова будинку» і виберіть **Type** – **Business Process (IDEF0)**. Натисніть кнопку **OK**.

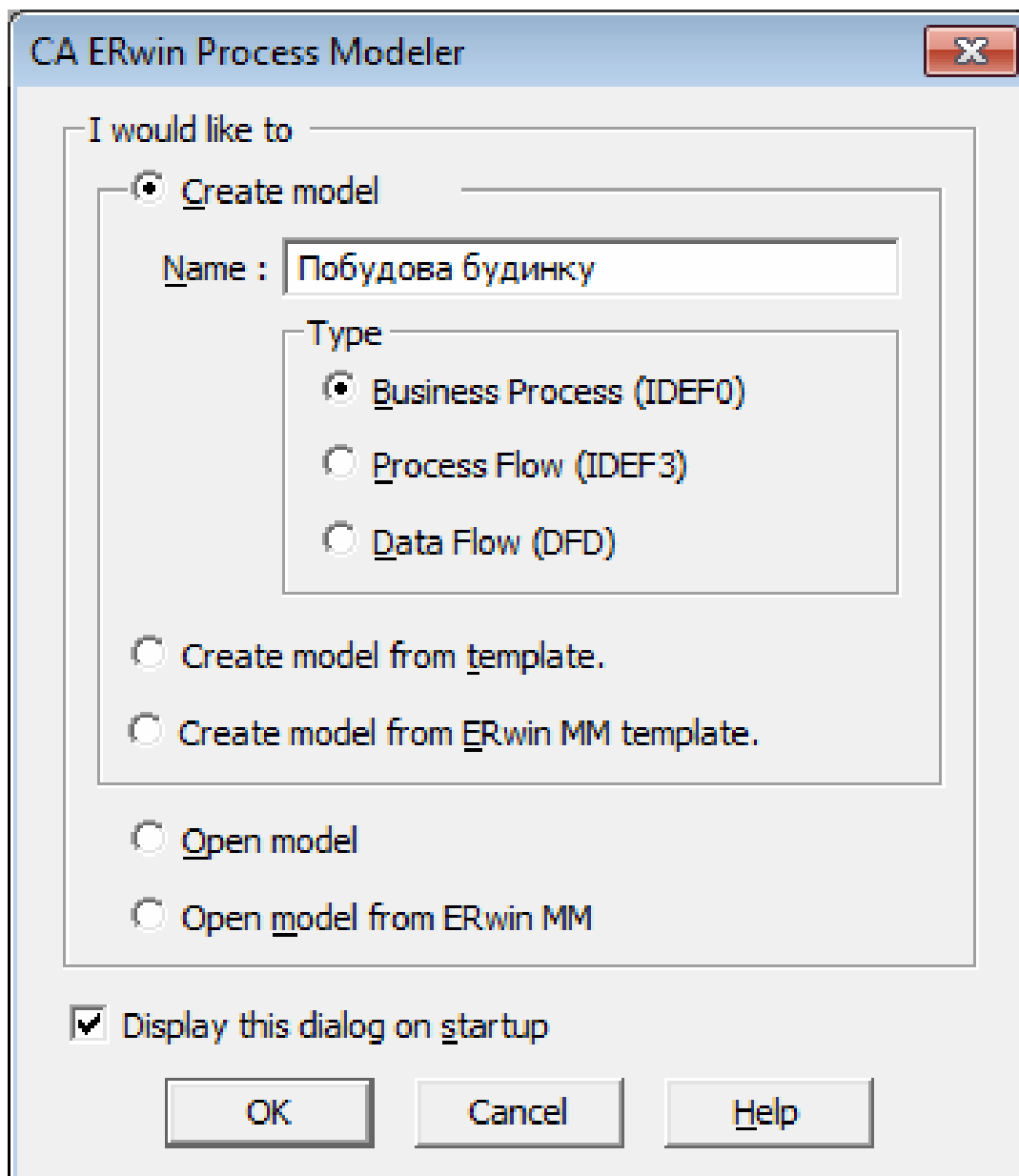


Рисунок 2 – Діалогове вікно **I would like to**

Відкриється діалогове вікно **Properties for New Models** (Властивості нової моделі), зображене на рисунку 3.

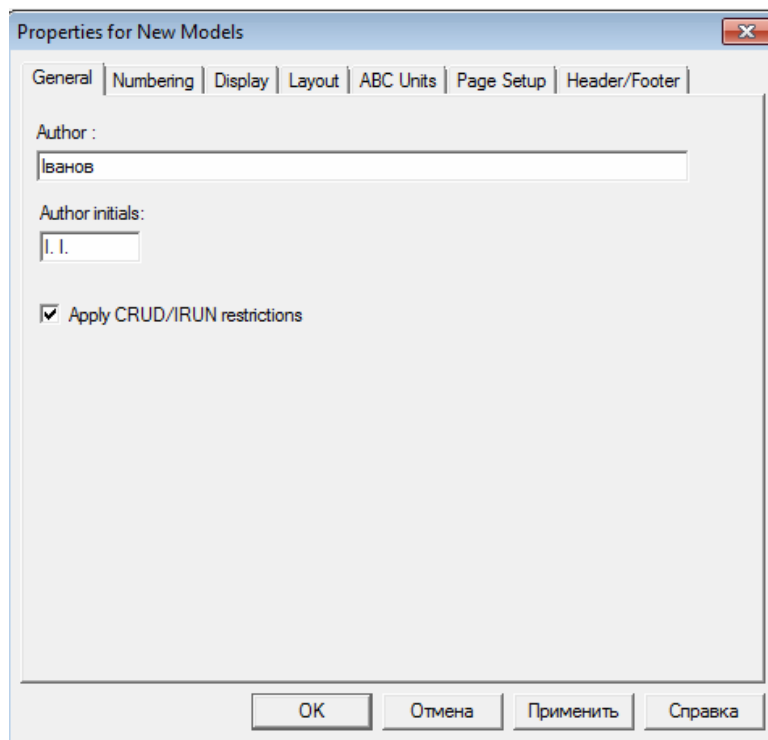


Рисунок 3 – Діалогове вікно **Properties for New Models**

Введіть у текстове поле **Author** (Автор) ім'я автора моделі й у текстове поле **Author initials** його ініціали. Натисніть послідовно кнопки **Apply** і **OK**. Автоматично створюється незаповнена контекстна діаграма (рис. 4).

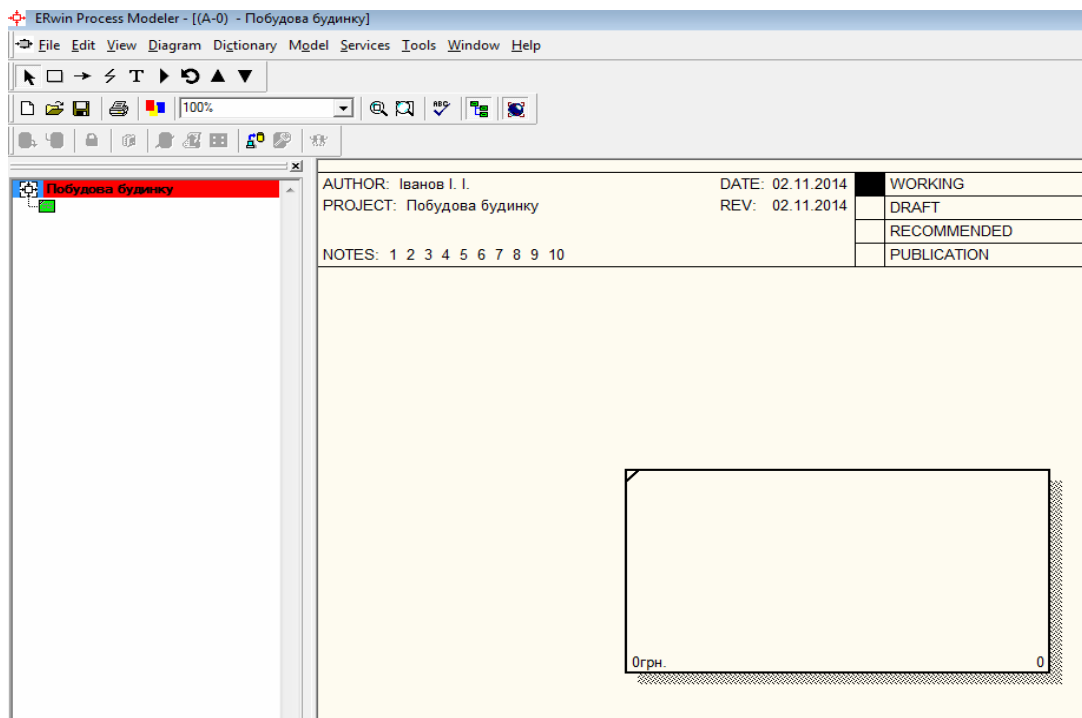


Рисунок 4 – Контекстна діаграма

У вкладці **Activities** клацання правою кнопкою по об'єкту в браузері моделі можна вибрати опції редагування його властивостей (рис. 5).

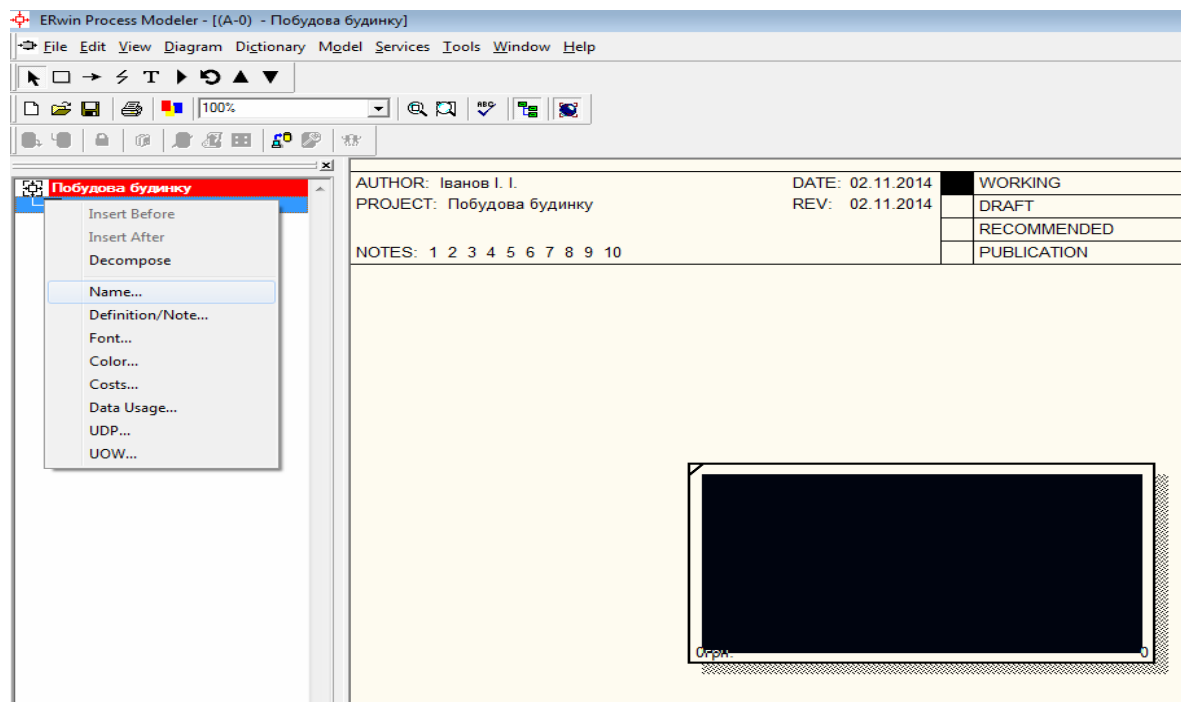


Рисунок 5 – Опція редагування властивостей об'єкту

Перейдіть у меню **Model/Model Properties**. У вкладці **General** діалогового вікна **Model Properties** (рис. 6) у текстові поля **Model name** слід внести ім'я моделі «Побудова будинку», **Project** ім'я проекту «Модель побудови будинку», **Author** (Автор) ім'я автора моделі, **Author initials** його ініціали, **Time Frame** (часове охоплення) – **AS-IS** (як є).

У вкладці **Purpose** діалогового вікна **Model Properties** (рис. 7) у текстове поле **Purpose** (мета) внесіть дані про мету розробки моделі – «Моделювати поточних (AS-IS) об'єктів та процесів будівельної компанії», а в текстове поле **Viewpoint** (точка зору) – «Головний інженер».

У вкладці **Definition** діалогового вікна **Model Properties** (рис. 8) у текстове поле **Definition** (визначення) внесіть «Це навчальна модель, що описує побудову будинку» і в текстове поле **Scope** (охоплення) – «Загальне керування побудовою будинку: підготовка, зведення дому, проведення внутрішніх та зовнішніх робіт, здача дому».

Model Properties

Layout | ABC Units | Page Setup | Header/Footer | Shapes | Draw Style

General | Purpose | Definition | Source | Status | Numbering | Display

Model name:
Побудова будинку

Project:
Модель побудови будинку

Author :
Іванов

Author initials:
І. І. ☒ Apply CRUD/IRUN restrictions

Time Frame
☒ AS-IS
☐ TO-BE

OK Отмена Применить Справка

Рисунок 6 – Вкладка **General** діалогового вікна **Model Properties**

Model Properties

Layout | ABC Units | Page Setup | Header/Footer | Shapes | Draw Style

General | Purpose | Definition | Source | Status | Numbering | Display

Model Name:
Побудова будинку

Purpose:
Модельовати поточних (AS-IS) об'єктів та процесів будівельної компанії

Viewpoint:
Головний інженер

OK Отмена Применить Справка

Рисунок 7 – Вкладка **Purpose** діалогового вікна **Model Properties**

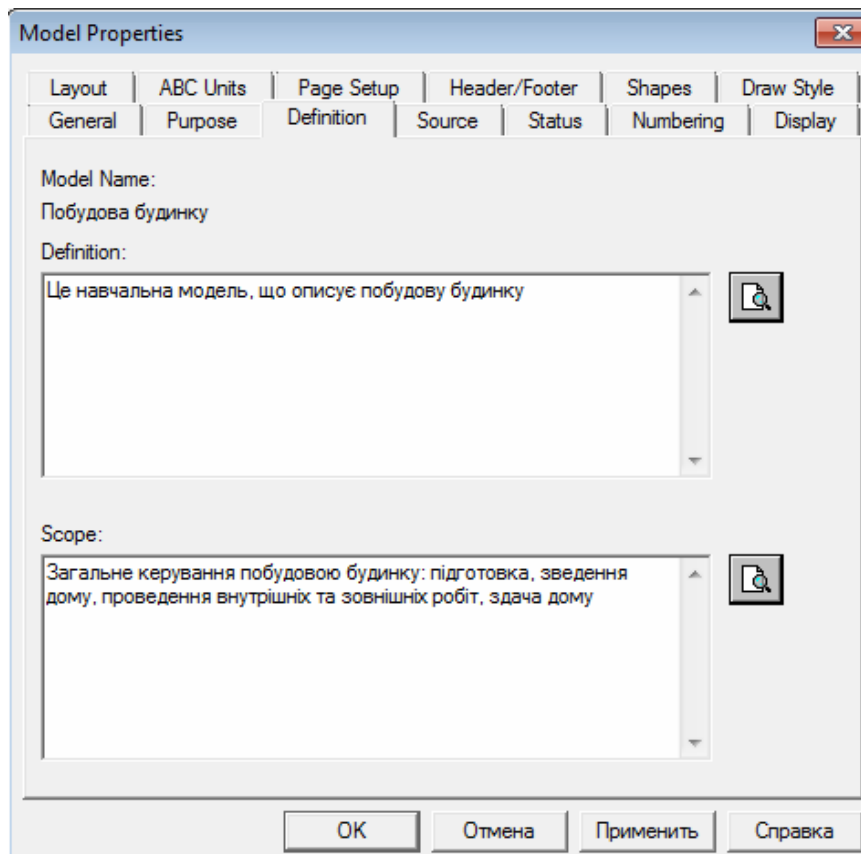


Рисунок 8 – Вкладка **Definition** діалогового вікна **Model Properties**

Перейдіть на контекстну діаграму й правою кнопкою миші клацніть по прямокутнику що представляє, у нотації **IDEF0**, умовне графічне позначення роботи. У контекстному меню виберіть опцію **Name** (рис. 9).

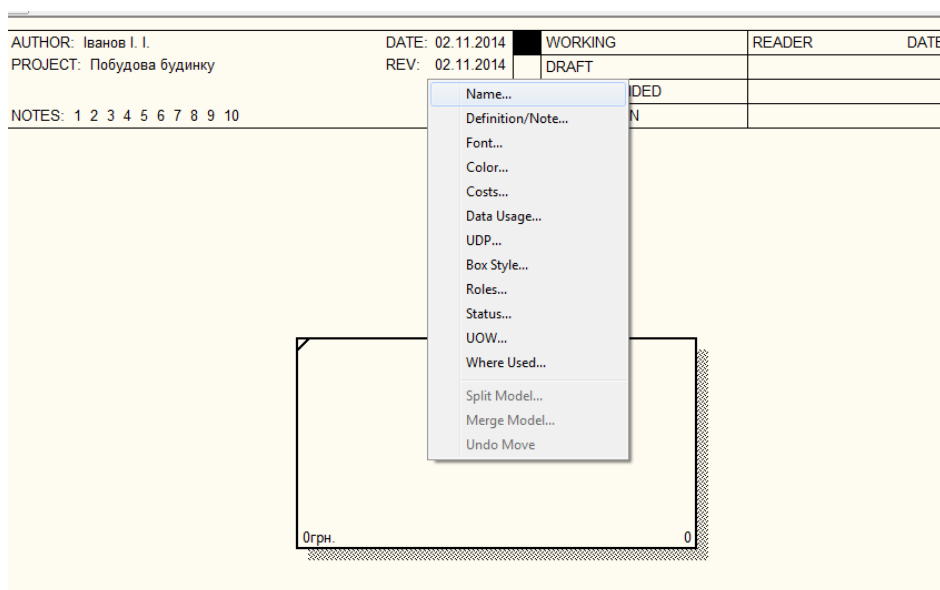


Рисунок 9 – Контекстне меню позначення роботи

У вкладці **Name** внесіть ім'я «Побудова будинку» (рис. 10). Якщо вам незрозуміло, як виконати ту чи іншу дію, ви можете викликати контекстну допомогу – клавіша **F1** або скористатися меню **Help**.

USED AT:	AUTHOR: Іванов І. І. PROJECT: Побудова будинку NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 02.11.2014 REV: 08.02.2015	<input checked="" type="checkbox"/> WORKING <input type="checkbox"/> DRAFT <input type="checkbox"/> RECOMMENDED <input type="checkbox"/> PUBLICATION	READER: _____ DATE: _____	CONTEXT: TOP
<div style="border: 1px solid black; padding: 10px; width: 200px; margin: auto;"> Побудова будинку 0грн. 0 </div>					
NODE: A-0	TITLE: Побудова будинку			NUMBER: <div style="border: 1px solid black; width: 50px; height: 20px; display: inline-block;"></div>	

Рисунок 10 – Назва роботи на контекстній діаграмі

Для того щоб текст став зрозумілий, в контекстному меню виберіть Font.

У діалоговому вікні **Activity Properties** в нижній частині вкладки Font встановіть прапорці в опціях **Apple setting to**, що дозволяють змінити шрифт для всіх робіт на поточній діаграмі, в моделі, і в групі **Global**, що дозволяє змінити шрифт одночасно для всіх об'єктів моделі, в опції **Script** виберіть «кириличний».

Встановіть шрифт **Arial Unicode MS**, звичайний, 14 пт (рис. 11).

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

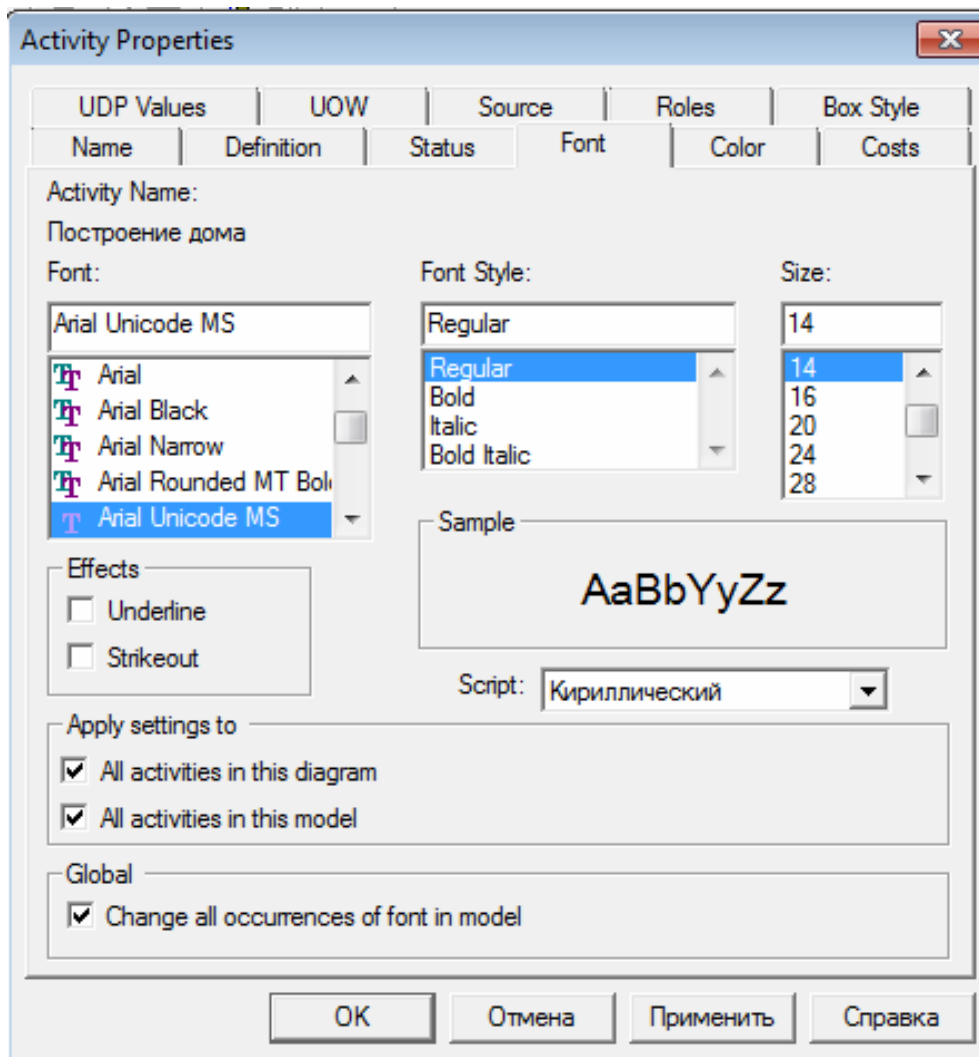


Рисунок 11 – Налаштування параметрів вкладки Font

Функціональність панелі інструментів (табл. 1, табл. 2) доступна з основного меню ERwin.

На основній панелі інструментів розташовані інструменти редактора ERwin для IDEF0-діаграм (рис. 12).

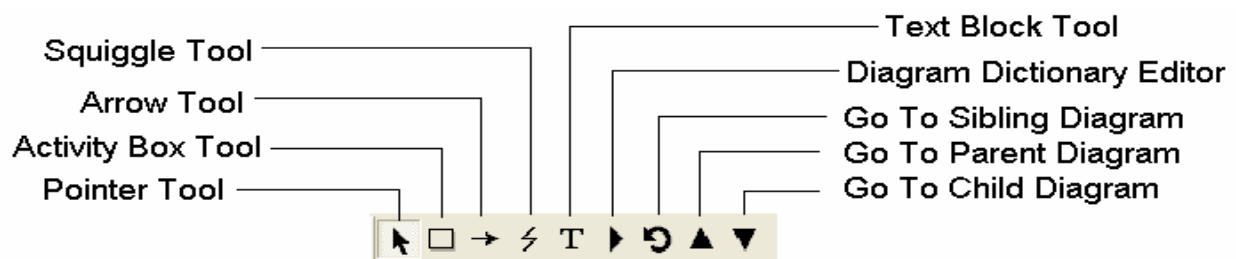


Рисунок 12 – Панелі інструментів редактора ERwin

Таблица 1 – Элементы управления основной панели инструментов ERwin






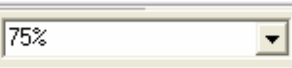




Элемент управления	Описание	Соответствующий пункт меню
	Создать новую модель	File/New
	Открыть модель	File/Open
	Сохранить модель	File/Save
	Напечатать модель	File/Print
	Вызвать генератор отчетов Report Builder	Tools/Report Builder
	Выбор масштаба	View/Zoom
	Масштабирование	View/Zoom
	Проверка правописания	Tools/Spelling
	Включение и выключение навигатора модели Model Explorer	View/Model Explorer
	Включение и выключение дополнительной панели инструментов работы с ModelMart	ModelMart

Таблица 2 – Инструменты редактора ERwin

Элемент редактирования	Описание	Название элемента
	Выбор и определение позиции объектов, добавленных в диаграмму.	Pointer Tool
	Установка блоков в диаграмме.	Activity Box Tool
	Установка дуги в диаграмме.	Arrow Tool
	Создание тильды (squiggle), которая соединяет дугу с ее названием.	Squiggle Tool
	Создание текстовых блоков.	Text Block Tool
	Отображение следующей диаграммы того же уровня	Go to Sibling Diagram
	Открытие диалогового окна Diagram Dictionary Editor, где можно перейти на какую-либо диаграмму или создать новую диаграмму.	Diagram Dictionary Editor
	Переход на родительскую диаграмму.	Go to Parent Diagram
	Отображение диаграммы потомка или разложение выделенного блока на диаграмму потомка	Go to Child Diagram

Далі необхідно створити **Стрілки** на контекстній діаграмі (табл. 3).

Таблиця 3 – Стрілки контекстної діаграми

Назва стрілки (Arrow Name)	Тип стрілки (Arrow Type)	Зміст стрілки
Технічне завдання Положення про охорону праці ДСТУ з будівництва Законодавчі акти	Control (управління)	Документація, згідно до якої виконують процес
Дозвіл на будівництво, Матеріали, Гроші	Input (вхід)	Інформаційні та матеріальні потоки
Побудований дім	Output (вихід)	Інформаційні та матеріальні потоки
Бригада будівельників Головний інженер Інвестор	Mechanism (механізми)	Люди або комп'ютерні системи, що виконують процес

Для побудови стрілок управління необхідно:

1. Вибрати на панелі інструментів кнопку.
2. Підвести курсор до верхнього краю вікна побудови діаграми до появи чорної смуги і зробити клацання лівою кнопкою миші по цій смугі (рис. 13).



Рисунок 13 – Початок побудови стрілки управління

3. Підвести курсор миші до верхньої сторони блока до утворення темного трикутника і клацнути лівою кнопкою миші (рис. 14).

Для побудови стрілок входу виконуються ті ж дії, але в такому порядку: від лівої сторони вікна побудови діаграми до лівого боку блока діаграми.

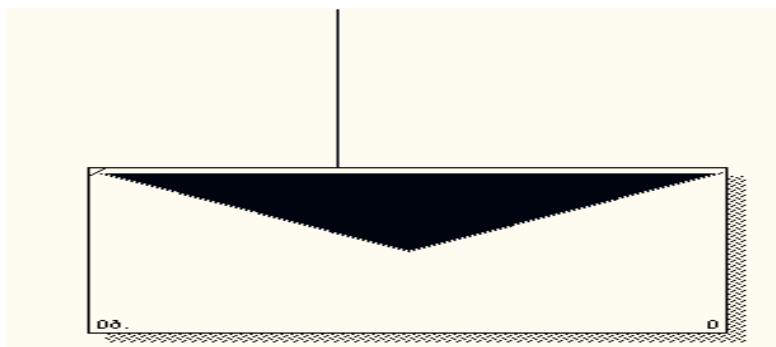



Рисунок 14 – Побудова стрілки управління

Для побудови стрілок виходу виконуються ті ж дії, але в такому порядку: від правого боку блока діаграми до правої сторони вікна побудови діаграм.

Для побудови стрілок механізмів виконуються ті ж дії, але в такому порядку: від нижнього краю вікна побудови діаграм до нижньої сторони блока діаграми.

Далі необхідно кожній стрілці дати назву:

1. Виберіть на панелі редагування кнопку .
2. Клацніть правою кнопкою миші по стрілці. Виберіть команду **Name**. Діаграма прийме вигляд, поданий на рисунку 15.

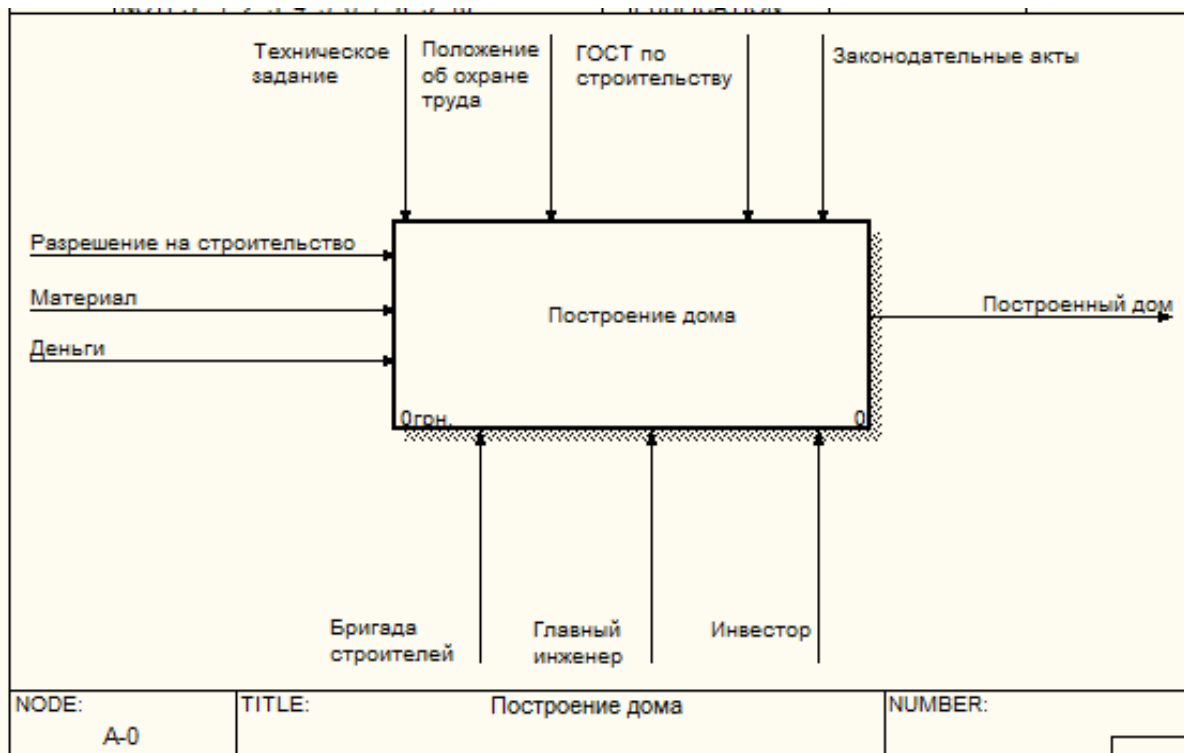


Рисунок 15 – Контекстна діаграма процесу «Побудова будинку» у нотації IDEF0

ERwin дозволяє встановити шрифт за замовчуванням для об'єктів певного типу на діаграмах. Після вибору меню **Model – Default Fonts** з'являється каскадне меню, кожен пункт якого служить для установки шрифтів для певного типу об'єктів (рис. 16).

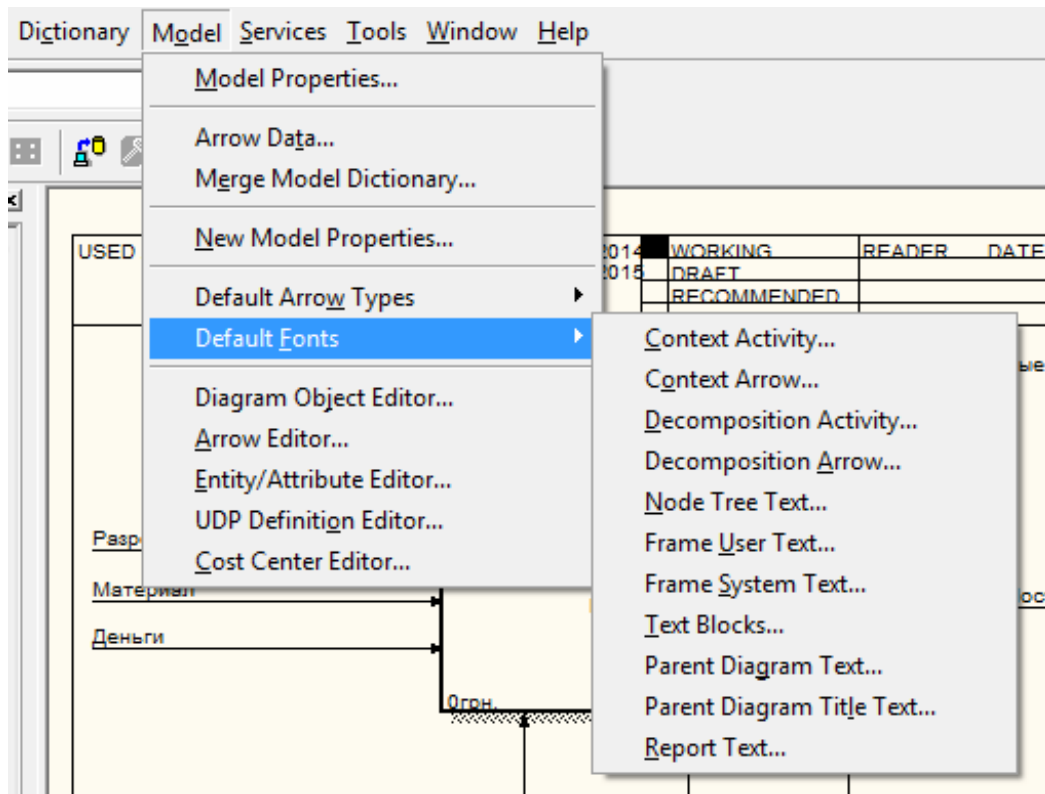


Рисунок 16 – Встановлення шрифтів за замовчуванням для об'єктів

Встановлюються шрифти за замовчуванням для таких об'єктів:

- **Context Activity** – робота на контекстній діаграмі;
- **Context Arrow** – стрілки на контекстній діаграмі;
- **Decomposition Activity** – роботи на діаграмі декомпозиції;
- **Decomposition Arrow** – стрілки на діаграмі декомпозиції;
- **Node Tree Text** – текст на діаграмі дерева вузлів;
- **Frame User Text** – текст, внесений користувачем у каркасі діаграм;
- **Frame System Text** – системний текст в каркасі діаграм;
- **Text Blocks** – текстові блоки;
- **Parent Diagram Text** – текст батьківської діаграми;
- **Parent Diagram Title Text** – текст заголовка батьківської діаграми;
- **Report Text** – текст звітів.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Для зміни кольору тексту виконайте команду контекстно-залежного меню **Color** (рис. 17).

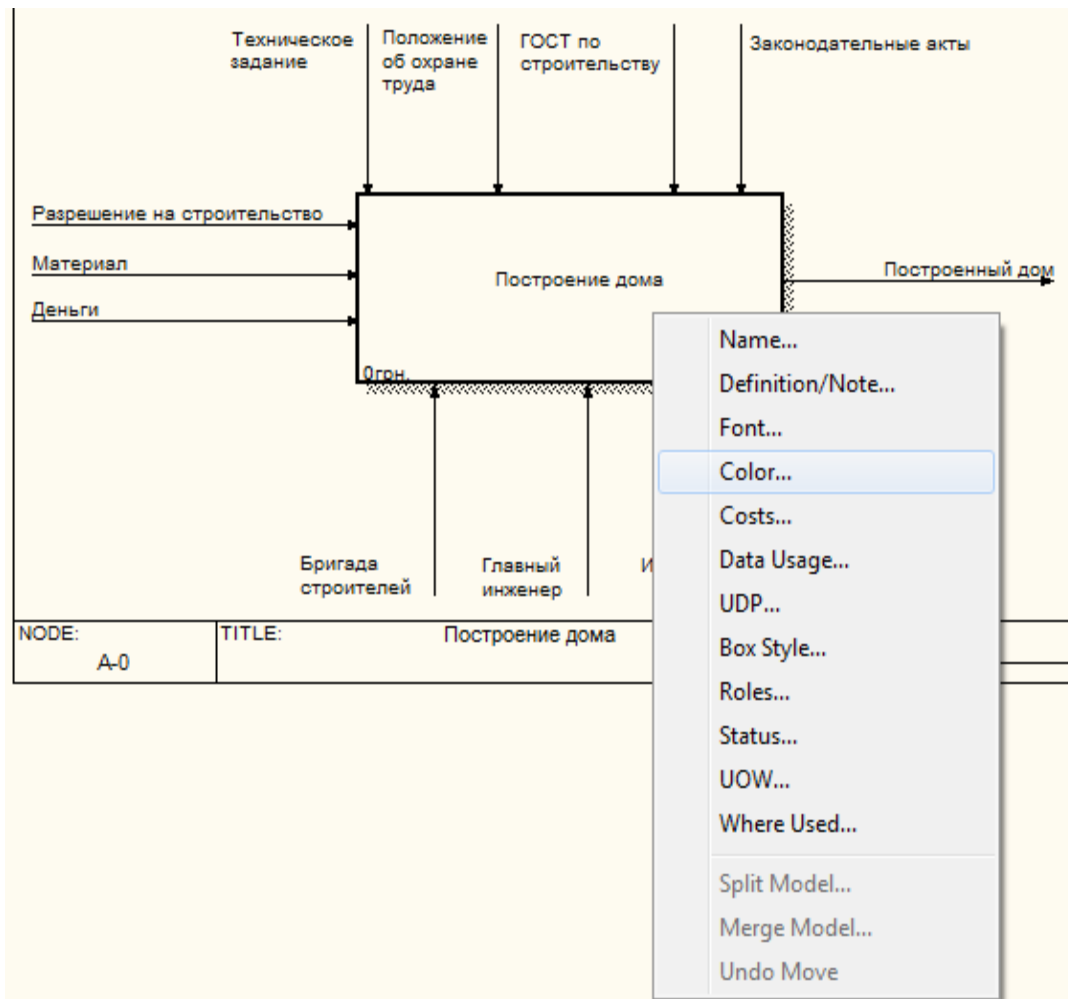


Рисунок 17 – Зміна кольору тексту та стрілок

Виберіть колір і натисніть кнопку  (рис. 18).

Для зміни фону блоку виберіть **Background Color** і колір (рис. 19).

Для зміни стилю стрілки виберіть у контекстно-залежному меню команду **Style** (рис. 20).

У діалоговому вікні вкажіть тип і стиль стрілки (рис. 21), натисніть на кнопку ОК.

Для видалення блоку і стрілки або тексту необхідно їх активізувати клацанням лівої кнопки миші і натиснути клавішу **Delete**, а потім підтвердити наміри з приводу видалення.

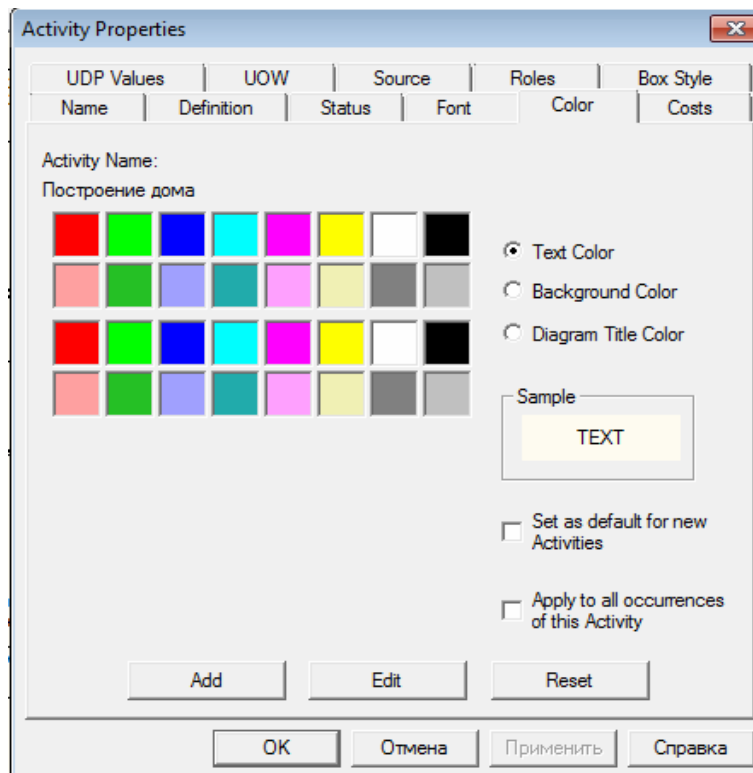


Рисунок 18 – Діалогове вікно вибору кольору тексту і стрілок

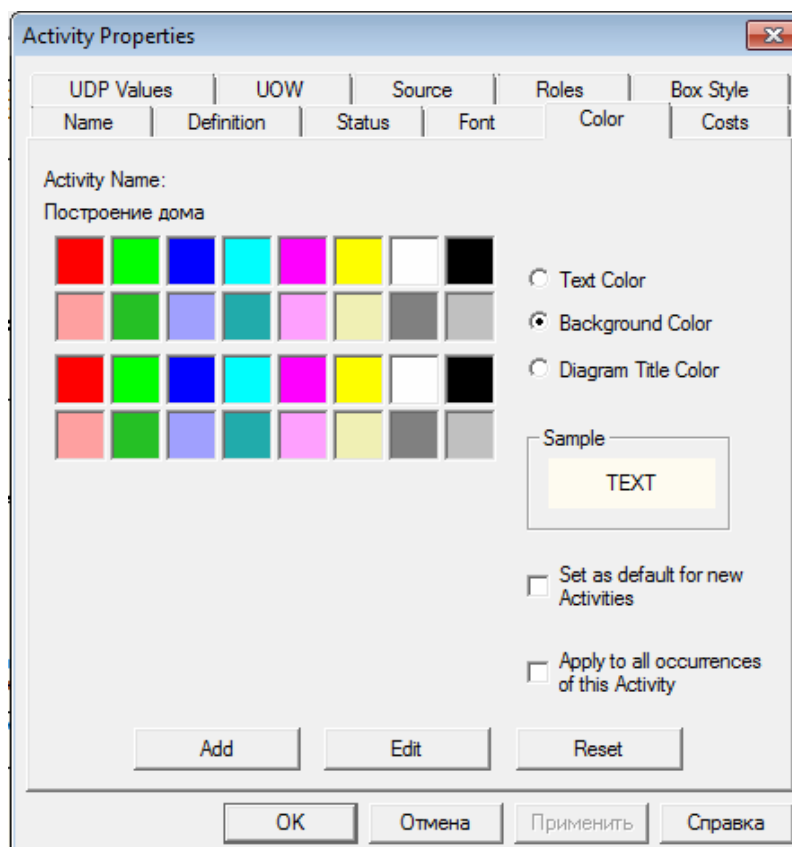


Рисунок 19 – Вкладка **Color** діалогу **Activity Properties**

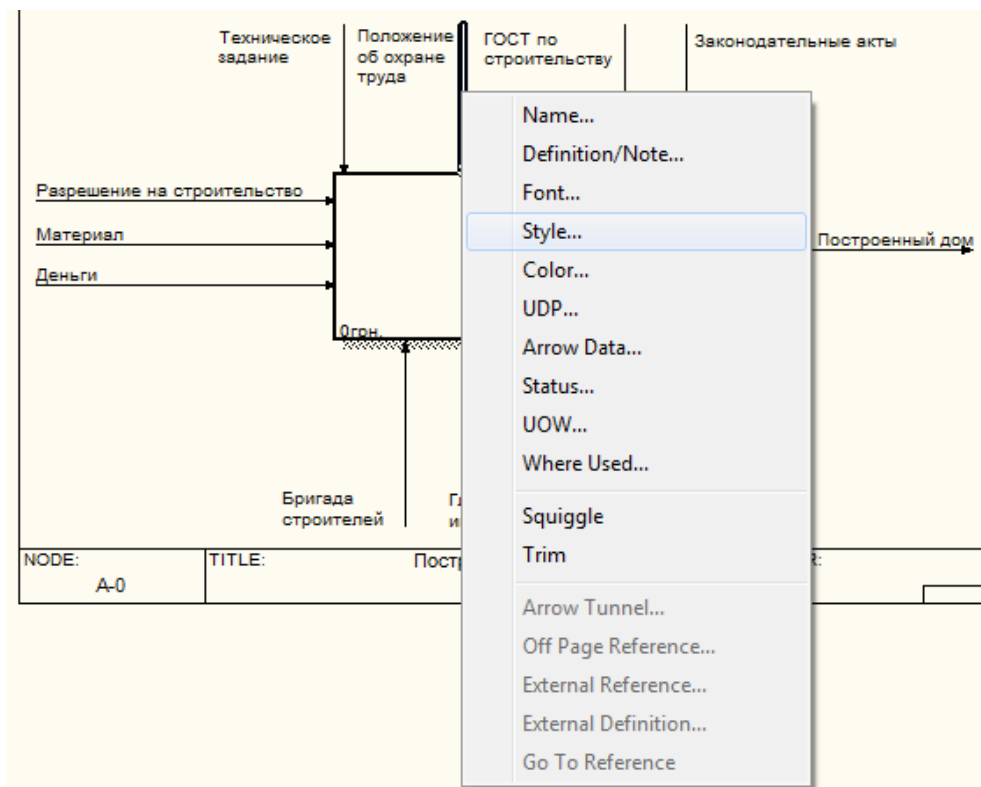


Рисунок 20 – Контекстно-залежне меню

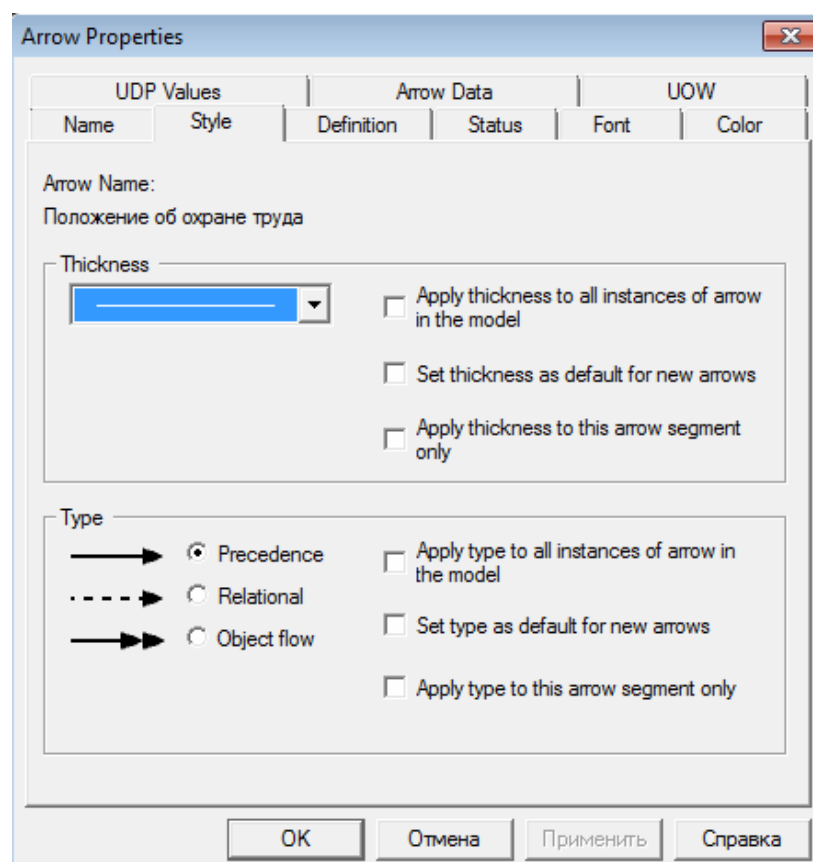


Рисунок 21 – Діалогове вікно **Arrow Properties**

Самостійно зробить форматування всіх елементів діаграми, спираючись на дані, наведені в таблиці 4.

Таблиця 4 – Таблиця редагування

Об'єкт	Дія	Порядок виконання
Блок	Зміна розміру	Для зміни висоти перетягнути мишею верхню чи нижню межу блоку, аналогічно змінюється розмір по горизонталі.
Текст	Стандартні способи форматування	Правою клавішею мишки клацнути по тексту, вибрати пункт контекстно-залежного меню Font Editor .
Стрілка	Зміна стилю, кольору, розміру	Натиснути правою клавішею мишки по стрілці і вибрати відповідний пункт: Style Editor , Color Editor або Trim .
Видалення блоків, стрілок або тексту: активізувати клацанням лівої кнопки мишки необхідний об'єкт і натиснути Delete , а потім підтвердити запит на видалення.		

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Збережіть отриману діаграму:

1. Створіть папку, назвіть її своїм прізвищем і в неї зберігайте свої роботи.
2. У меню **File** виберіть команду **Save as**.
3. Вкажіть шлях до своєї папки і ім'я файлу **Lab1.bp1**.
4. Натисніть на кнопку **ЗБЕРЕГТИ**.

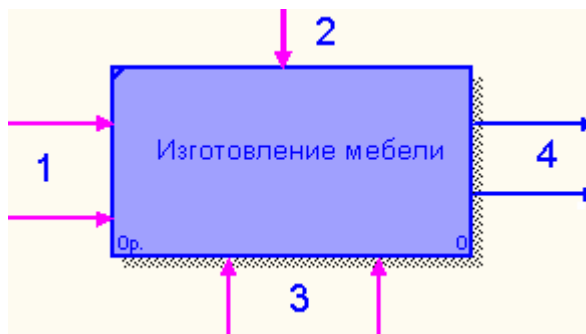
Після того, як викладач визначить Ваш варіант роботи, починайте його виконання. Це завдання ви будете робити протягом усіх практичних робіт, поступово вивчаючи програмне середовище ERwin та відпрацьовуючи навички роботи в ньому.

Після того, як робота буде виконана, запросіть викладача і продемонструйте йому результат.

Контрольні питання

Дайте відповідь на контрольні питання:

1. Перелічіть основні можливості ВРwin.
2. Охарактеризуйте основні елементи робочого інтерфейсу ВРwin.
3. Яку методологію підтримує ВРwin?
4. Вкажіть призначення кожної із стрілок, зображених на рисунку:



5. Назвіть основні етапи побудови моделі.
6. Який процес можна назвати функціональною декомпозицією?
7. Перерахуйте елементи контекстної діаграми.
8. За допомогою якого інструменту будуються стрілки на діаграмі?

Тестові завдання

1. Модель об'єктів та процесів – це:

- а) комплекс діаграм, кожна з яких описує окремий об'єкт та процес;
- б) ієрархія діаграм, кожна з яких описує окремий об'єкт та процес;
- в) випадковий набір діаграм, кожна з яких описує окремий об'єкт та процес;
- г) всі відповіді правильні;
- д) правильної відповіді немає.

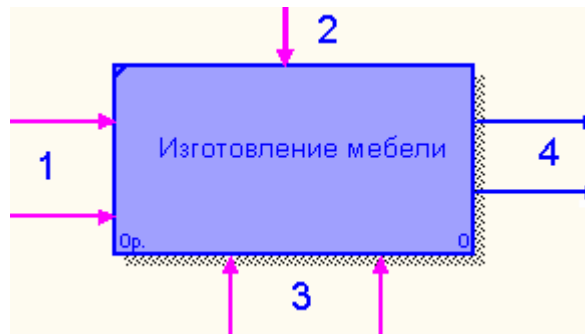
2. Що входить до складу діаграм:

- а) блоки, що описують підпроцеси (функції системи), і стрілки, що зв'язують блоки разом і зображують взаємодії та взаємозв'язки між блоками;
- б) блоки, що описують підпроцеси (функції системи);
- в) стрілки, що зв'язують блоки разом і зображують взаємодії і взаємозв'язки між блоками;
- г) нумерація діаграми;
- д) правильні відповіді б), г).

3. Вкажіть етап, в якому відбувається побудова діаграми процесу верхнього рівня:

- а) малювання стрілки управління;
- б) визначення основного об'єкту та процесу;
- в) побудова контекстної діаграми;
- г) функціональна декомпозиція кожного процесу;
- д) всі відповіді правильні.

4. Вкажіть, для чого призначена ліва сторона блоку (1):



- а) для управління (правила, стратегії, стандарти);
- б) для механізмів (ресурси, які виконують процес);
- в) для виходів (матеріал або інформація, що одержані в результаті виконання процесу);
- г) для входів (матеріал або інформація, що використовуються або перетворюються для отримання результату);
- д) правильні відповіді в), г).

5. Яка команда з контекстно-залежного меню дозволить змінити стиль стрілки:

- а) Font Editor;
- б) Color Editor;
- в) Trim;
- г) Style Edition;
- д) Background Color;
- е) Squiggle.

6. Для чого призначений інструмент **T**:

- а) для створення тильди (squiggle), яка з'єднує стрілку з її назвою;
- б) для вибору і визначення позиції об'єктів, доданих в діаграму;
- в) використовується для створення текстових блоків;
- г) використовується для вибору кольору тексту і фону;
- д) правильної відповіді немає.

1.1.2 Декомпозиція контекстної діаграми

Завдання: придумати і реалізувати в інструментальному програмному засобі моделювання VFPwin свій приклад процесу.

Обмеження. Контекстна діаграма повинна містити всі типи граничних стрілок (вхід, вихід, управління та механізм). Декомпозиція контекстної діаграми повинна містити не менше 4-х дій. Кожна дія в діаграмі декомпозиції контекстної діаграми має бути декомпозованою на не менше, ніж чотири дії. Діаграми декомпозицій повинні містити всі типи внутрішніх стрілок.

Після створення контекстної діаграми, яка є описом контексту модельованої системи (процесу), проводиться функціональна декомпозиція – система (процес) розбивається на підсистеми (підпроцеси) і кожна підсистема (підпроцес) описується в тому ж синтаксисі, що і система (процес) в цілому. Потім кожна підсистема (підпроцес) розбивається на більш дрібні і так до досягнення потрібного рівня деталізації. В результаті такого розбиття, кожен фрагмент системи (процесу) зображується на окремій діаграмі декомпозиції. Діаграма декомпозиції призначена для деталізації роботи.

Під час декомпозиції процесу всі стрілки, вхідні або вихідні з нього, повинні бути перенесені на діаграму нижнього рівня і використані під час її побудови. При цьому заборонені нові стрілки, що виходять за межі нової діаграми стрілок. Діаграма верхнього рівня створюється шляхом декомпозиції основного процесу контекстної діаграми. На діаграмі декомпозиції процеси нумеруються автоматично зліва направо. Номер процесу показується в правому нижньому кутку.


Деталізація процесу «Побудова будинку», створення діаграми A0.

Відкрийте файл Lab1.bp1, збережений на попередньому уроці.

Наступним кроком є деталізація контекстного процесу за допомогою діаграми верхнього рівня. Ця діаграма містить в собі чотири процеси:

- а) Процес 1.1 – Підготовка до будівництва будинку.
- б) Процес 1.2 – Зведення будинку.
- в) Процес 1.3 – Проведення внутрішніх та зовнішніх робіт.
- г) Процес 1.4 – Здача дому.

Проведіть деталізацію процесу «Побудова будинку», задавши потрібну кількість нових блоків. Для цього:

1. Клацніть по блоку «Побудова будинку» і виберіть інструмент .
2. У діалоговому вікні введіть число (рекомендовано від 4 до 8), на яке буде проведена декомпозиція (наприклад, 4).

3. Вкажіть тип діаграми **IDEF0** (рис. 22) і натисніть **OK**.

4. Вкажіть назви нових блоків («Підготовка до будівництва будинку», «Зведення будинку», «Проведення внутрішніх та зовнішніх робіт», «Здача дому»).

Під час декомпозиції процесу ті стрілки, що входили до нього і виходили з нього автоматично з'являються на діаграмі декомпозиції (міграція дуг).

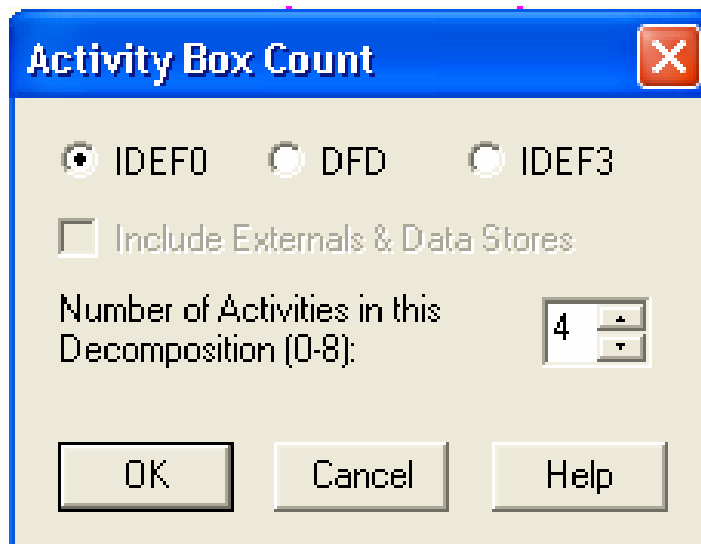


Рисунок 22 – Діалогове вікно декомпозиції блоку

Такі стрілки називаються непов'язаними і сприймаються в ERwin, як синтаксична помилка. Визначимо вхідні і вихідні потоки для нових процесів.

Зв'яжемо мігруючі дуги:

1. Виберіть інструмент малювання стрілок .

2. Клацніть мишею по наконечнику вхідного потоку Дозвіл на будівництво, Матеріал, Гроші.

3. Клацніть по вхідній стороні блоку «Підготовка до будівництва будинку».

4. Клацніть мишею по наконечнику вихідного потоку Побудований дім.

5. Клацніть по вихідній стороні блоку «Здача дому».

Малювання стрілок між блоками:

1. Виберіть інструмент малювання стрілок .

2. Клацніть лівою кнопкою миші по вихідній стороні попереднього блоку, а потім по вхідній стороні наступного блоку.

3. Аналогічно підведіть стрілки управління та механізмів.

4. Перевірте себе (рис. 23).

Збереження отриманої діаграми:

1. У меню **File** виберіть **Save as**.
2. Вкажіть шлях до своєї папки і ім'я файлу **Lab2.bp1**.
3. Натисніть **ОК**.

Після того, як робота буде виконана, запросіть викладача і продемонструйте йому результат.

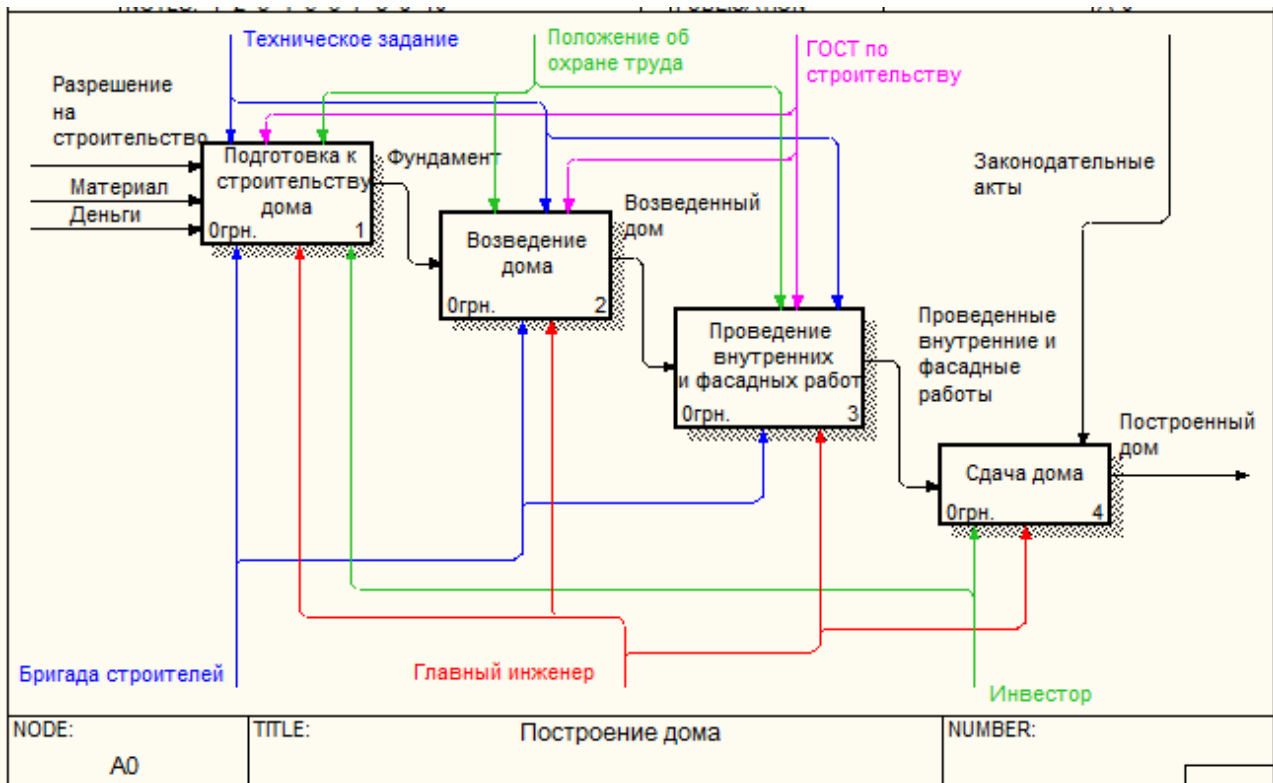


Рисунок 23 – Діаграма декомпозиції першого рівня процесу «Побудова будинку»

Контрольні питання

1. Як створюється діаграма верхнього рівня?
2. Як на діаграмі відображається декомпозиція?
3. Яким інструментом задається кількість блоків для декомпозиції?
4. Які стрілки ERwin сприймає як синтаксичну помилку?

Тестове завдання

1. У якому порядку потрібно розташовувати блоки на діаграмі деталізації:

- а) в довільному;
- б) обов'язково справа наліво;
- в) відповідно до рівня складності подальшої деталізації;
- г) відповідно до їх домінуванням;
- д) всі відповіді правильні.






2. Для чого призначена діаграма деталізації процесу:

- а) для позначення стрілок;
- б) для опису контексту модельованої системи;
- в) для деталізації роботи;
- г) всі відповіді правильні.






3. Як створюється діаграма верхнього рівня:

- а) шляхом декомпозиції однієї з функцій контекстної діаграми;
- б) шляхом декомпозиції основної функції контекстної діаграми;
- в) шляхом декомпозиції діаграм нижнього рівня;
- г) шляхом декомпозиції однієї з функцій на діаграмі верхнього рівня;
- д) правильної відповіді немає.

4. Який інструмент необхідно вибрати, щоб з його допомогою зробити декомпозицію блоку?

- а) ; б) ; в) ; г) ; д) .

5. Яким інструментом проводиться процес зв'язування мігруючих дуг?

- а) ; б) ; в) ; г) ; д) .

6. Мігруюча дуга – це дуга, яка ...

- а) створює зворотний зв'язок з управління;
- б) переходить з батьківської діаграми і не стосується блоку;
- в) переходить з батьківської діаграми і стосується блоку;
- г) створює зворотний зв'язок по входу;
- д) правильної відповіді немає.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

1.1.3 Діаграма декомпозиції другого рівня

Останнім кроком побудови моделі є функціональна декомпозиція. Побудована діаграма верхнього рівня також має множину процесів, які в свою чергу можуть бути деталізовані в діаграми нижнього рівня.

Таким чином, будується ієрархія IDEF0 з контекстною діаграмою в вершині ієрархії.

Цей процес декомпозиції триває до досягнення потрібного рівня деталізації. Кожний процес більш низького рівня необхідно співвіднести з процесом верхнього рівня. Зазвичай, для цієї мети всі роботи моделі нумеруються. Номер складається з префікса і числа. Може бути використаний префікс будь-якої довжини, наприклад, А.

Контекстна робота дерева має номер А0. Роботи декомпозиції А0 мають номери А1, А2, А3 і т. д. Роботи декомпозиції нижнього рівня мають номер батьківської роботи і черговий порядковий номер, наприклад, роботи декомпозиції А3 будуть мати номери А31, А32, А33, А34 і т. д.

Роботи утворюють ієрархію, де кожна робота може мати одну батьківську і кілька дочірніх робіт, утворюючи дерево. Таке дерево називають **деревом вузлів**, а вищеописану нумерацію – **нумерацією по вузлах**.

На попередніх практичних роботах Ви побудували контекстну діаграму процесу «Побудова будинку» і провели його деталізацію за допомогою діаграми верхнього рівня. Останнім кроком побудови моделі є функціональна декомпозиція, тобто розбиття складних процесів на більш прості. Цей процес декомпозиції триває до досягнення потрібного рівня деталізації.

Використовуючи кроки надані в 1.1.2, побудуйте чотири діаграми декомпозиції другого рівня (рис. 24 – рис. 27).

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

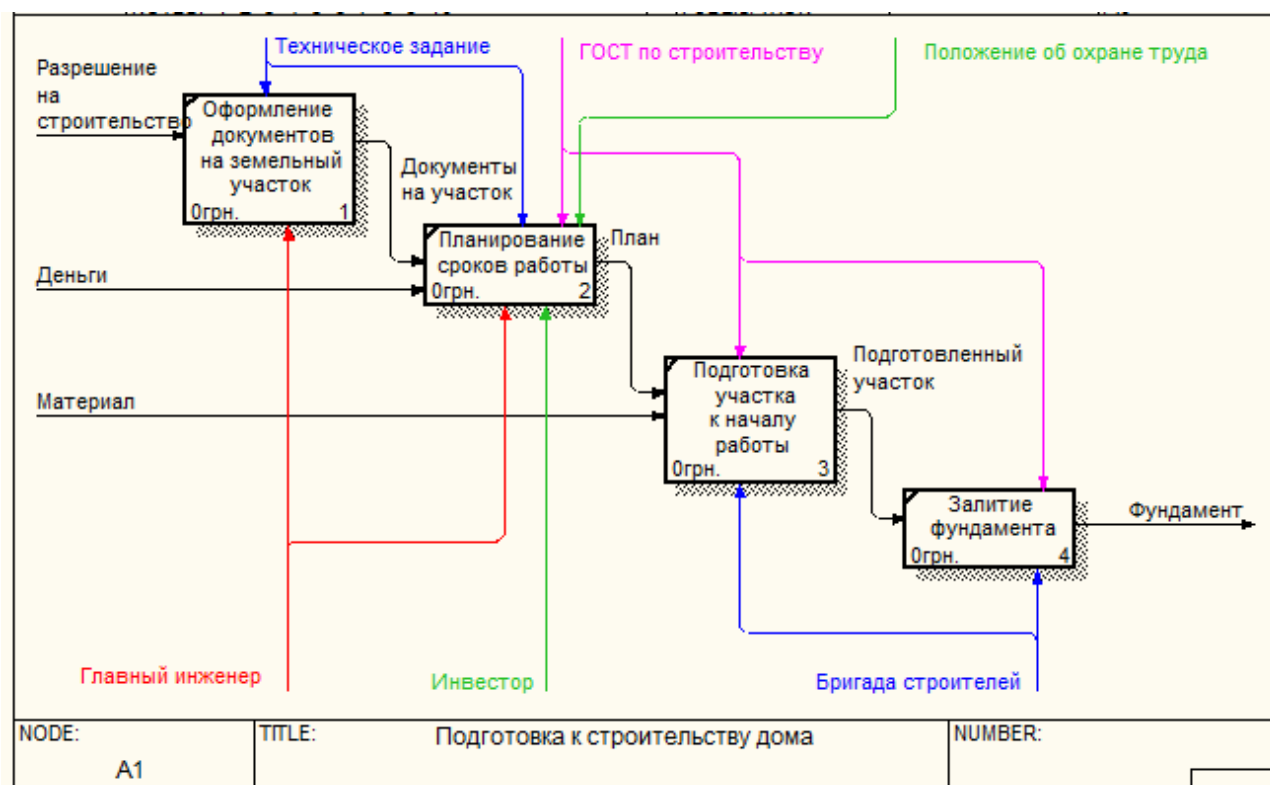


Рисунок 24 – Діаграма декомпозиції другого рівня процесу «Підготовка до будівництва дому»

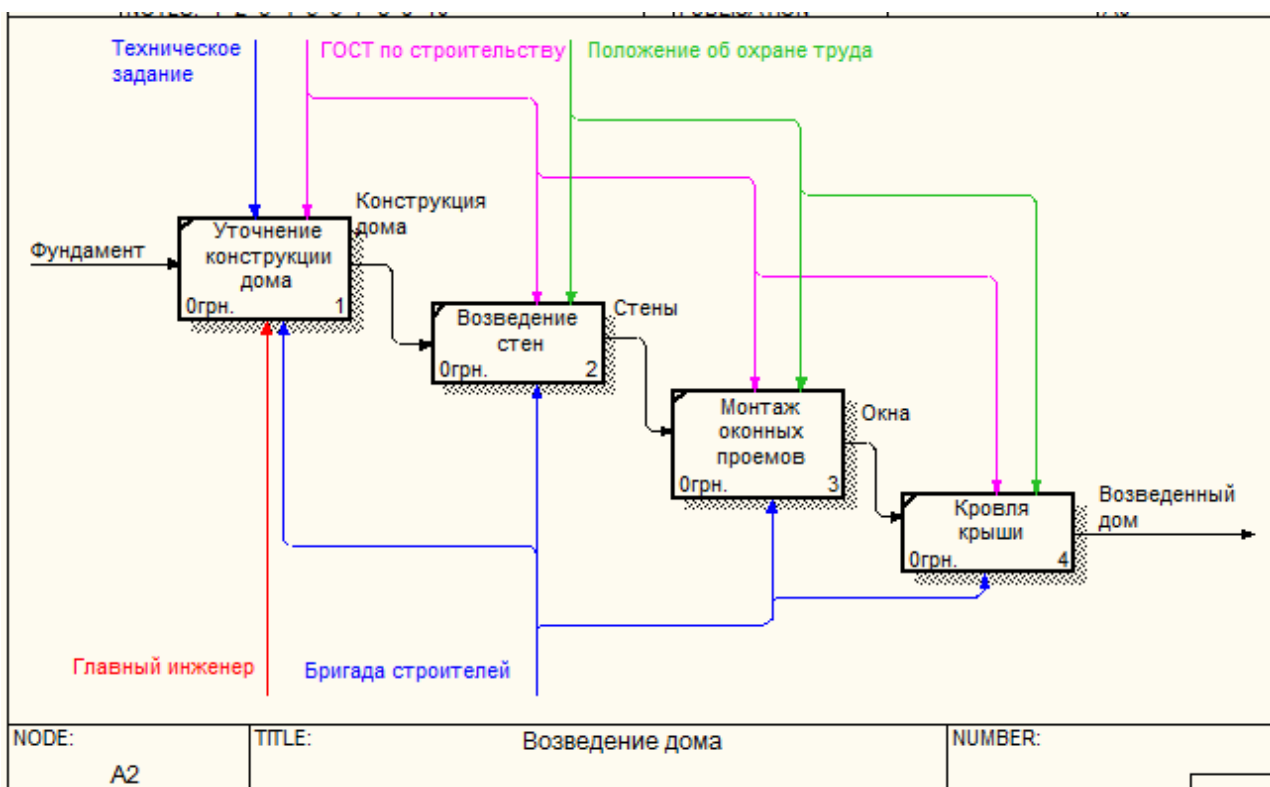


Рисунок 25 – Діаграма декомпозиції другого рівня процесу «Зведення дому»

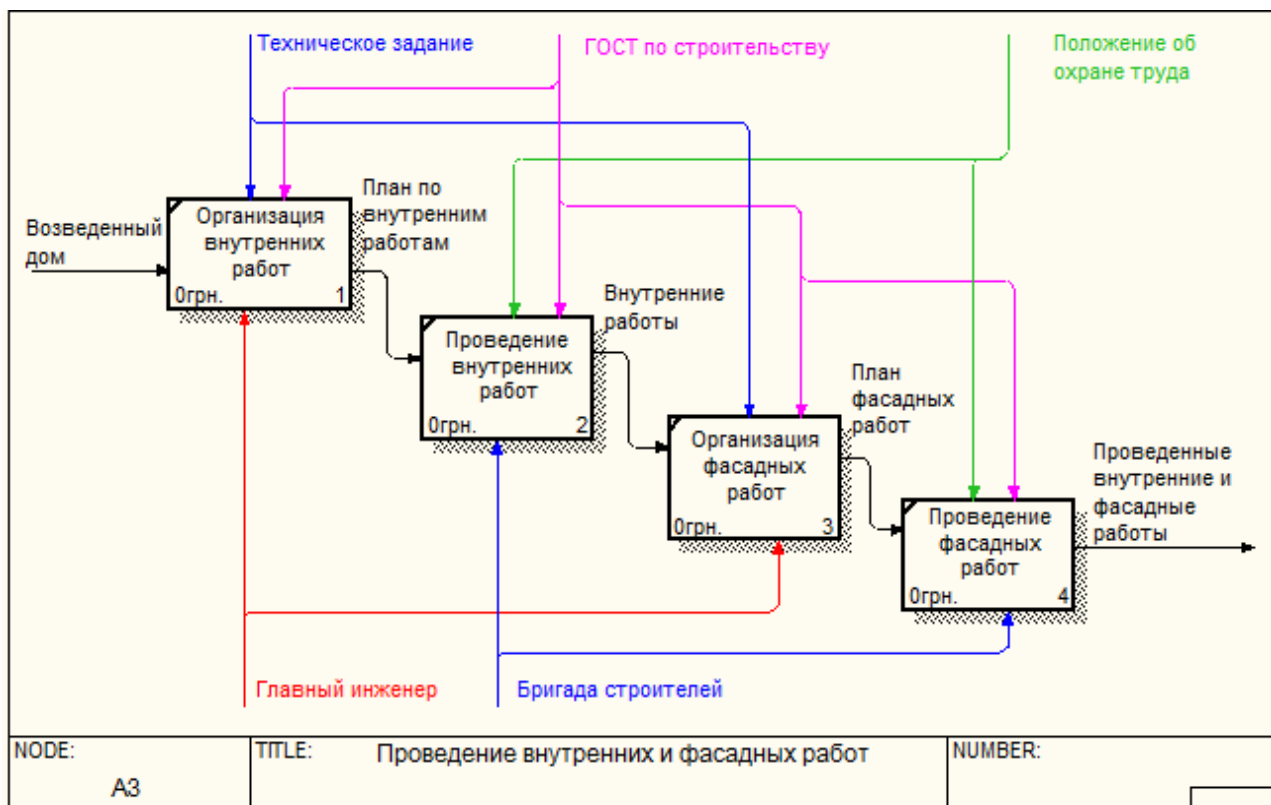


Рисунок 26 – Диаграмма декомпозиции второго уровня
процессу «Проведения внутренних та фасадных работ»

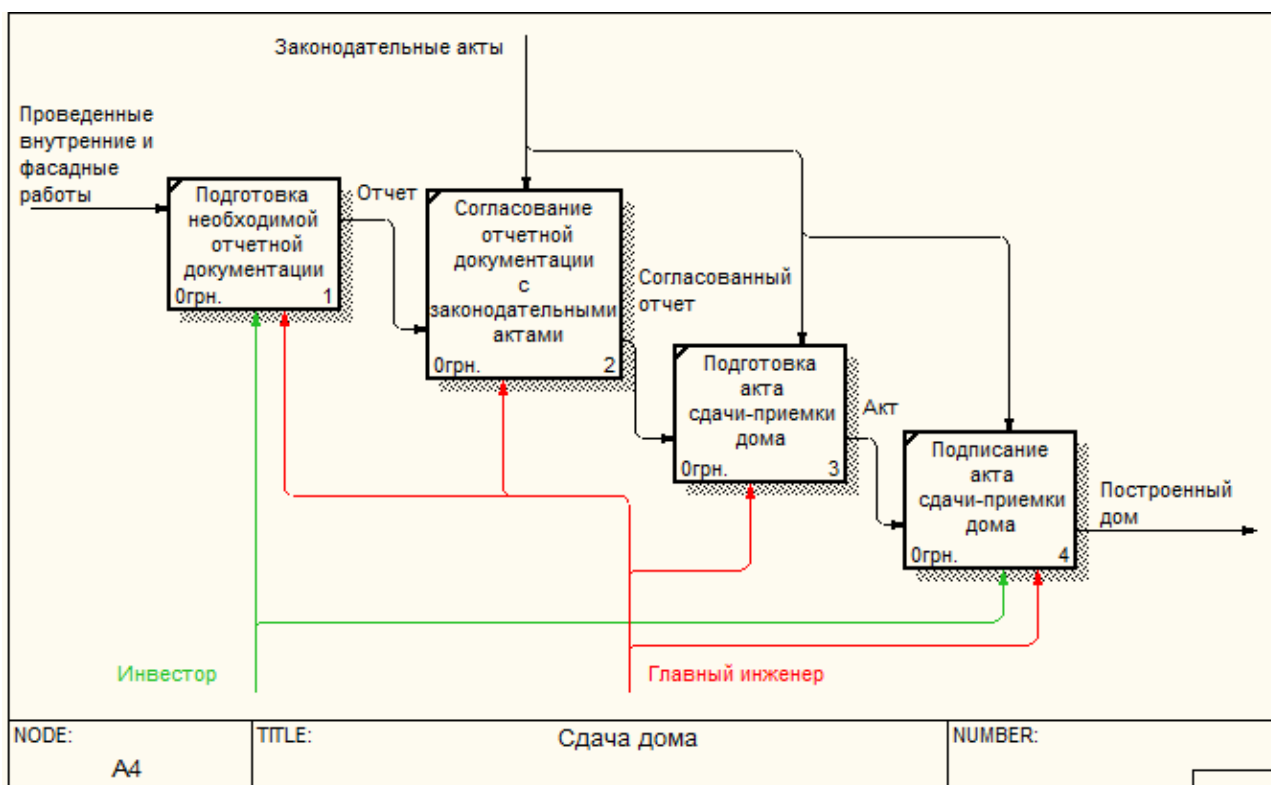


Рисунок 27 – Диаграмма декомпозиции второго уровня процессу «Здача дому»

Складання звіту.

Результат опису моделі можна отримати в звіті **Model Report**:

1. Діалогове вікно налаштування звіту по моделі викличте з пункту меню **Tools – Reports – Model Report**.
2. Виберіть необхідні поля, при цьому автоматично відображається черговість виведення інформації у звіті (рис. 28);
3. Натисніть на кнопку **Preview**, щоб переглянути звіт.

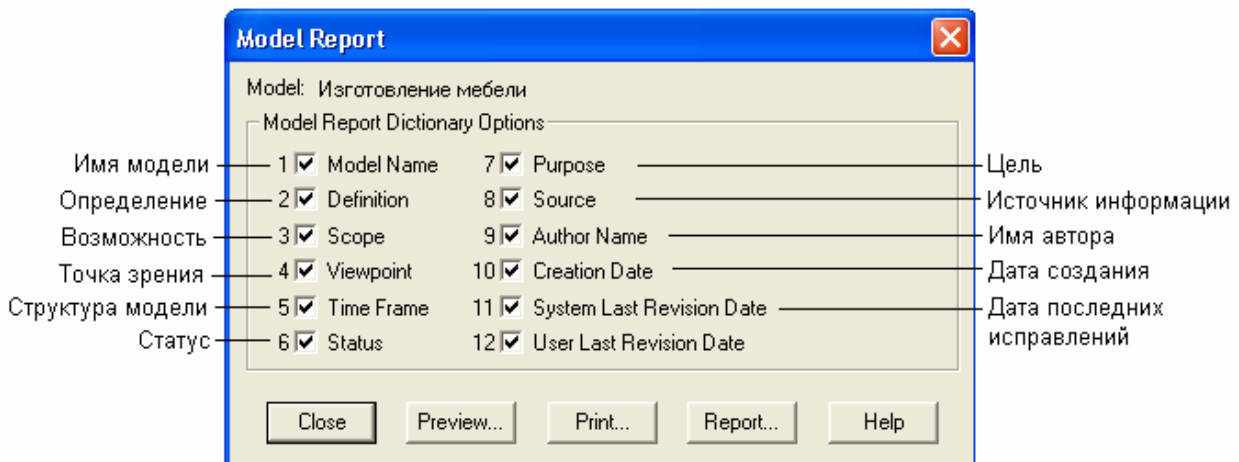


Рисунок 28 – Діалогове вікно вибору інформації для звіту

Збереження отриманих діаграм:

1. У меню **File** виберіть **Save As**.
2. Вкажіть шлях до своєї папки і ім'я файлу **Lab3.bp1**.
3. Натисніть **OK**.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Вказівки з оформлення звіту:

1. Звіт оформлюється у текстовому редакторі Microsoft Word.
2. Звіт повинен містити:
 - назву лабораторної роботи;
 - мета лабораторної роботи;
 - формулювання завдань;
 - короткий опис ходу виконання завдань;
 - діаграми, що відображають результати виконання завдань;
 - висновки по роботі.

Контрольні питання

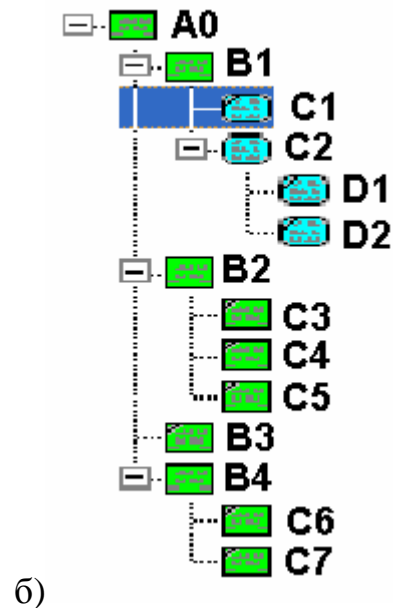
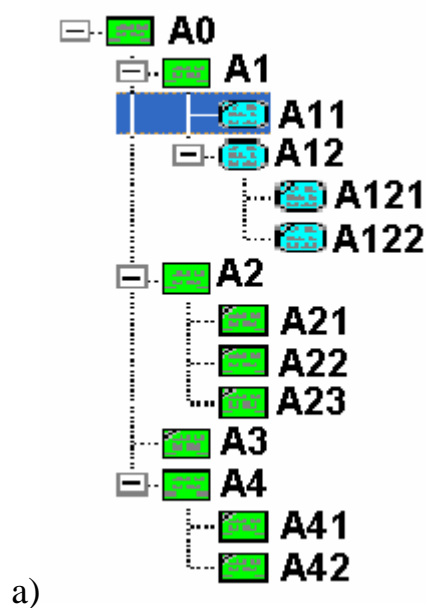
1. Як нумеруються моделі в ієрархії IDEF0?
2. Дайте визначення поняттю «Дерево вузлів».
3. Який процес розробки моделі є функціональною декомпозицією?
4. Як можна винести мету і точку зору проекту на діаграму?
5. Для чого необхідне складання звіту?

Тестове завдання

1. IDFE0 – це:

- а) діаграма потоку даних;
- б) діаграма бізнес-процесу;
- в) діаграма сутність-зв'язок;
- г) діаграма ключів;
- д) всі відповіді правильні.

2. Вкажіть номер рисунка, який відображає правильну нумерацію по вузлах:








3. Скільки контекстних діаграм може бути в моделі:

- а) одна;
- б) дві;
- в) стільки, скільки рівнів деталізації;
- г) необмежену кількість;
- д) всі відповіді правильні.

4. Дуги управління вказують на:

- а) готовий результат;
- б) проміжний результат;
- в) ресурси, які виконують процес;
- г) правила, стандарти;
- д) інформацію для отримання результату;
- е) всі відповіді правильні.

5. Який інструмент дозволяє внести мету і точку зору на контекстну діаграму?

- а) ; б) ; в) ; г) ; д) .

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

1.2 Використання інструментальних засобів для моделювання потоків даних

Мета роботи: ознайомлення з функціями інструментального програмного засобу моделювання об'єктів і процесів BPwin (ERwin), придбання навичок використання методології DFD для моделювання потоків даних.

Методичні вказівки по організації роботи студентів

Діаграми потоків даних (Data Flow Diagramming, DFD) можна використовувати як доповнення до моделі IDEF0 для більш наочного відображення поточних операцій у системах обробки інформації.

Діаграми потоків даних використовуються для опису обробки інформації та модельної системи як мережі пов'язаних між собою робіт. Діаграми потоків даних (DFD) показують зовнішні джерела та приймачі даних, потоки даних і сховища (накопичувачі) даних, до яких здійснюється доступ.

DFD описує:

- функції обробки інформації (роботи);
- документи (стрілки, **Arrows**), об'єкти, співробітників або відділи, які беруть участь в обробці інформації;
- зовнішні посилання (**External references**), які забезпечують інтерфейс з зовнішніми об'єктами, що знаходяться за межами модельованої системи;
- таблиці для зберігання документів (сховища даних, **Data store**).

Для зображення діаграм потоків даних в BRwin використовується нотація Гейна-Сарсона (рис. 29).

компонента	нотація Гейна-Сарсона
поток данных	ИМЯ →
управляющий процесс	<div> <div>НОМЕР</div> <div>ИМЯ</div> </div>
хранилище данных	<div> <div>№</div> <div>ИМЯ</div> </div>
внешняя сущность	<div> <div>НОМЕР</div> <div>ИМЯ</div> </div>

Рисунок 29 – Нотація Гейна-Сарсона

Етапи побудови моделі:

1. Виділення зовнішніх об'єктів, з якими система повинна бути пов'язана.
2. Формування DFD діаграми першого рівня.
3. Функціональна декомпозиція кожного підпроцесу за допомогою діаграм нижнього рівня.
4. Складання словника даних.
5. Побудова специфікацій процесу, якщо його не можна виразити комбінацією підпроцесів.

При доповненні моделі IDEF0 діаграмою DFD в палітрі інструментів на новій діаграмі DFD з'являються нові кнопки:



– додати в діаграму зовнішнє посилання (**External Reference**). Зовнішнє посилання є джерелом або приймачем даних ззовні моделі;



– додати в діаграму сховище даних (**Data store**). Сховище даних дозволяє описати дані, які необхідно зберегти в пам'яті перш, ніж використовувати в роботах.

На відміну від IDEF0, де система розглядається як взаємопов'язані роботи, DFD розглядає систему як сукупність предметів.

Роботи в DFD – функції системи, що перетворюють входи у виходи. Хоча роботи зображуються прямокутниками з округленими кутами, сенс їх збігається зі змістом робіт в IDEF0, вони мають входи і виходи, але не підтримують управління та механізми, як IDEF0 (рис. 30).

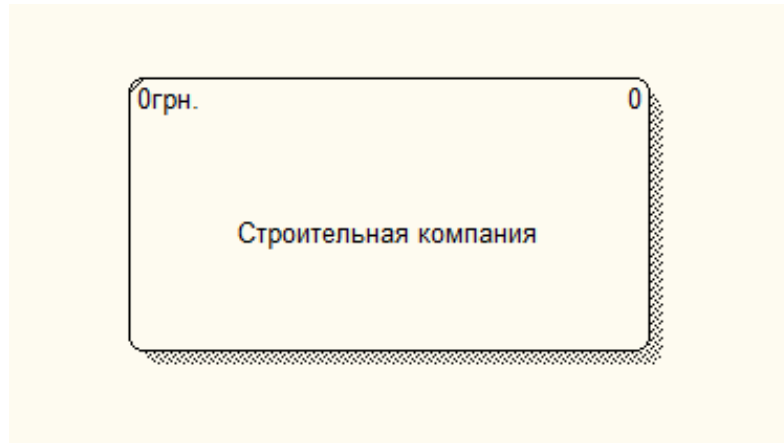


Рисунок 30 – Зображення системи або процесу

Зовнішні сутності зображують входи в систему і / або виходи із системи. Зовнішні сутності зображуються у вигляді прямокутника з тінню і, зазвичай, розташовуються по краях діаграми (рис. 31). Одна зовнішня сутність може бути використана багаторазово на одній або декількох діаграмах, такий прийом застосовують, щоб не малювати занадто довгих і заплутаних стрілок.



Рисунок 31 – Зовнішня сутність

Стрілки (потоки даних) описують рух об'єктів з однієї частини системи в іншу. Оскільки в DFD кожна сторона роботи не має чіткого призначення, як в IDEF0, стрілки можуть підходити і виходити з будь-якої грані прямокутника роботи. У DFD також застосовуються двонаправлені стрілки для опису діалогів типу команди-відповіді між роботами, між роботою і зовнішньою сутністю і між зовнішніми сутностями (рис. 32).

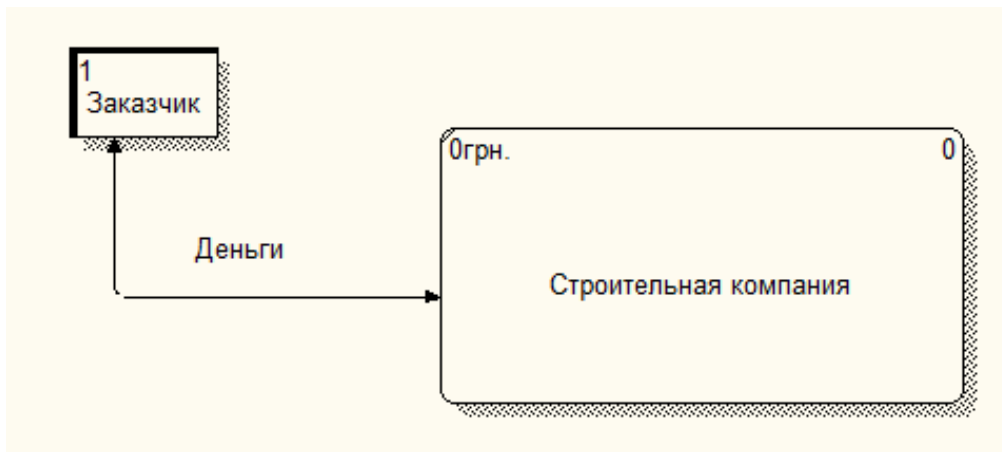


Рисунок 32 – Поток даних (стрілки)

Сховище даних зображує об'єкти в спокої. У матеріальних системах сховища даних зображуються там, де об'єкти очікують обробки, наприклад в черзі.

У системах обробки інформації сховища даних є механізмом, який дозволяє зберегти дані для подальших процесів (рис. 33).

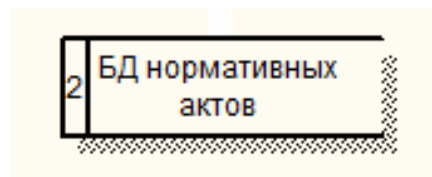


Рисунок 33 – Сховище даних

На відміну від стрілок IDEF0, які показують жорсткий взаємозв'язок, стрілки DFD представляють, як об'єкти (включаючи дані) рухаються від однієї роботи до іншої.

Подання потоків спільно з сховищами даних і зовнішніми сутностями робить моделі DFD більш схожими на фізичні характеристики системи – рух об'єктів (**Data flow**), зберігання об'єктів (**Data stores**), постачання і поширення об'єктів (**External entities**) (рис. 34).

Створення діаграми DFD.

1. Відкрийте файл **Lab3.bp1**, збережений на попередньому уроці.
2. Виберіть в діалозі **CA ERwin Process Modeler** радіо-кнопку **Data Flow** (DFD) (рис. 35).
3. Внесіть назву моделі «Побудова будинку» і клацніть по кнопці **OK**.

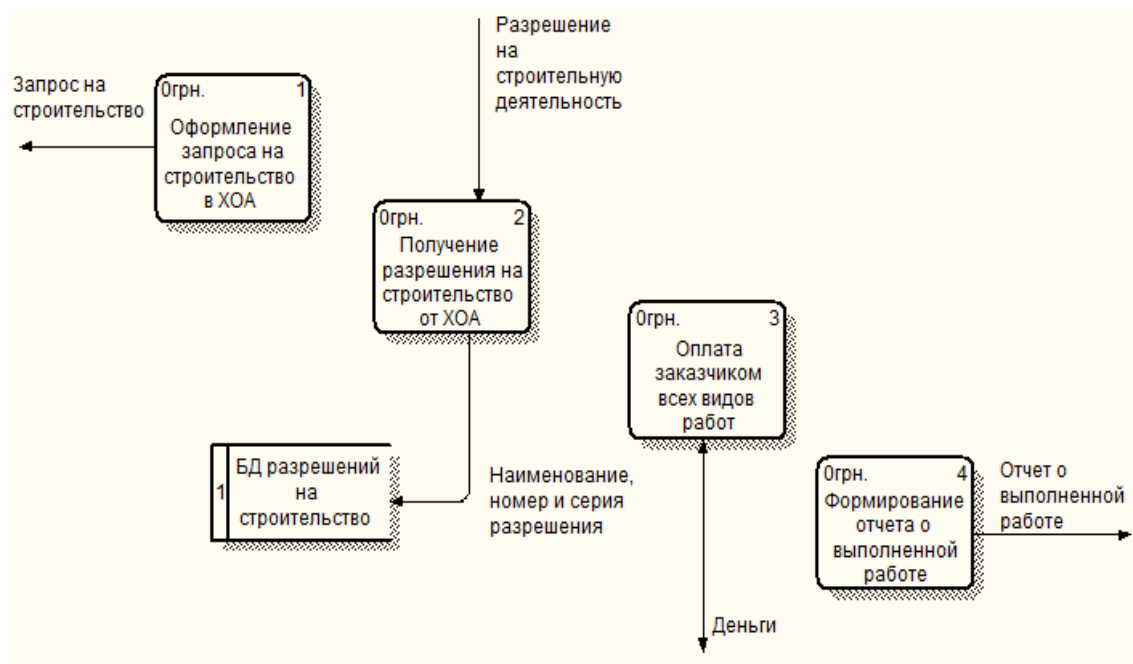


Рисунок 34 – Приклад діаграми DFD

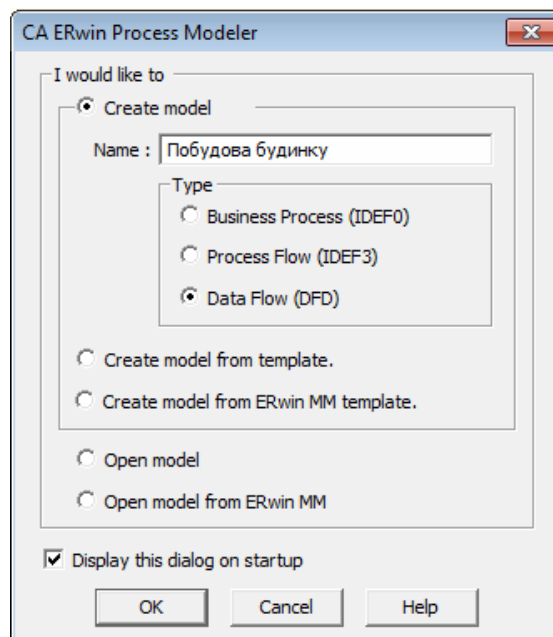


Рисунок 35 – Діалог CA Erwin Process Modeler

Зображення зовнішніх сутностей.

Виконайте так дії:

1. Використовуючи кнопку , внесіть зовнішні сутності (зовнішні посилання).

2. У діалоговому вікні введіть назву сутності «Замовник» (рис. 36) та натисніть **ОК**.

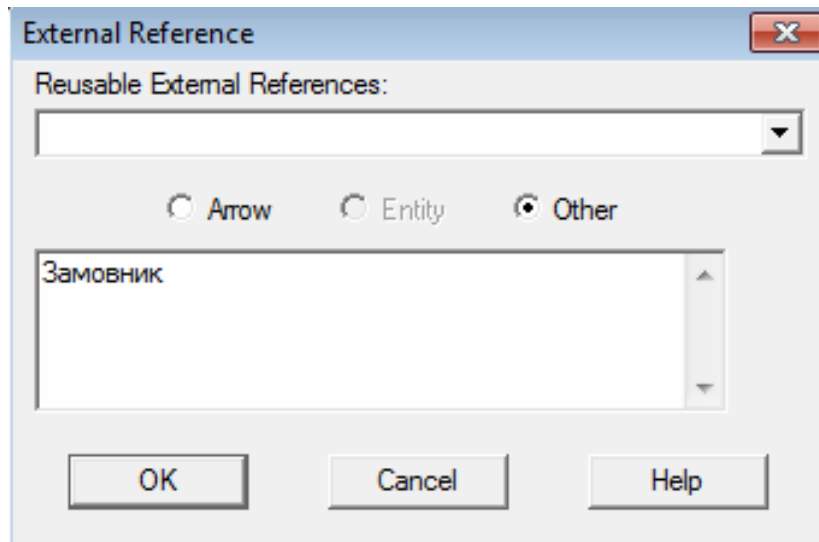



Рисунок 36 – Діалогове вікно зовнішньої сутності

Побудова сховища.

Використовуючи кнопку  на палітрі інструментів, внесіть сховище даних: «БД дозволів на будівництво» (рис. 37).

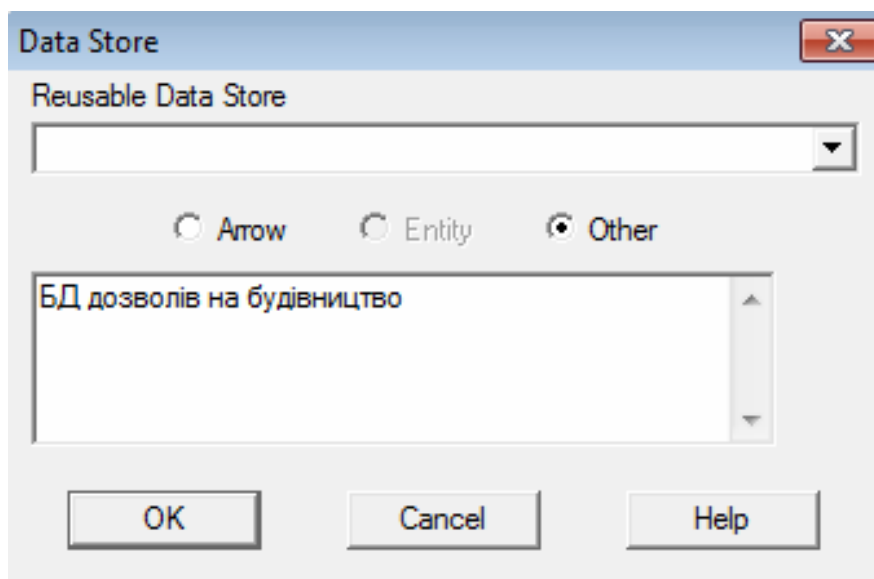



Рисунок 37 – Побудова сховища «БД дозволів на будівництво»

Створення внутрішніх посилань.

1. Використовуючи інструмент , створіть внутрішні посилання: «Гроші», «Дозвіл на будівельну діяльність», «Запит на будівництво», «Звіт про виконану роботу».

2. Стрілку «Гроші» необхідно зробити двонаправленою, для цього клацніть правою кнопкою по стрілці, виберіть у контекстному меню **Style**.

3. Виберіть у діалоговому вікні **Arrow Properties** (рис. 38) опцію **Bidirectional**.

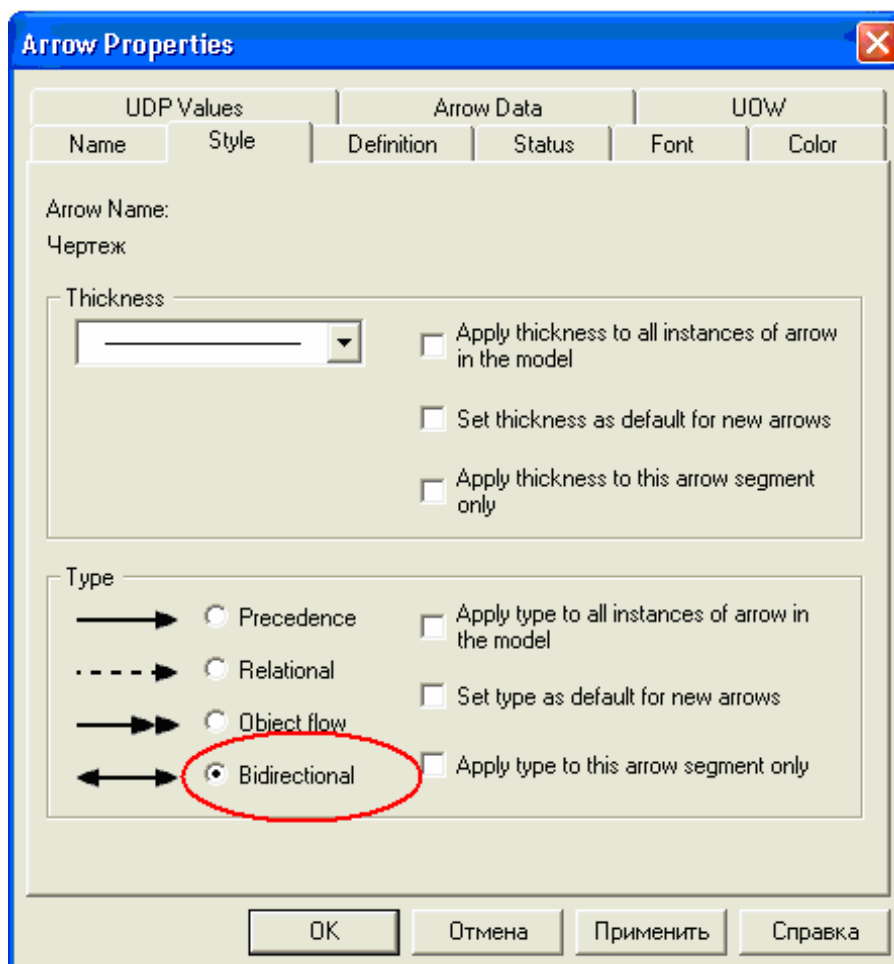


Рисунок 38 – Діалогове вікно вибору стилю стрілки

Завдання: придумати і реалізувати в інструментальному програмному засобі моделювання об'єктів і процесів BPwin (ERwin) свій приклад процесу в методології DFD.

Обмеження. Контекстна діаграма повинна містити зовнішнє посилання. Декомпозиція контекстної діаграми повинна містити не менше 4-х дій. Кожна дія в діаграмі декомпозиції контекстної діаграми має бути декомпозованою на не менше, ніж чотири дії. Діаграми декомпозицій повинні містити сховище даних.

Після виконання завдання у вас повинні вийти такі діаграми (рис. 39 – рис. 44).

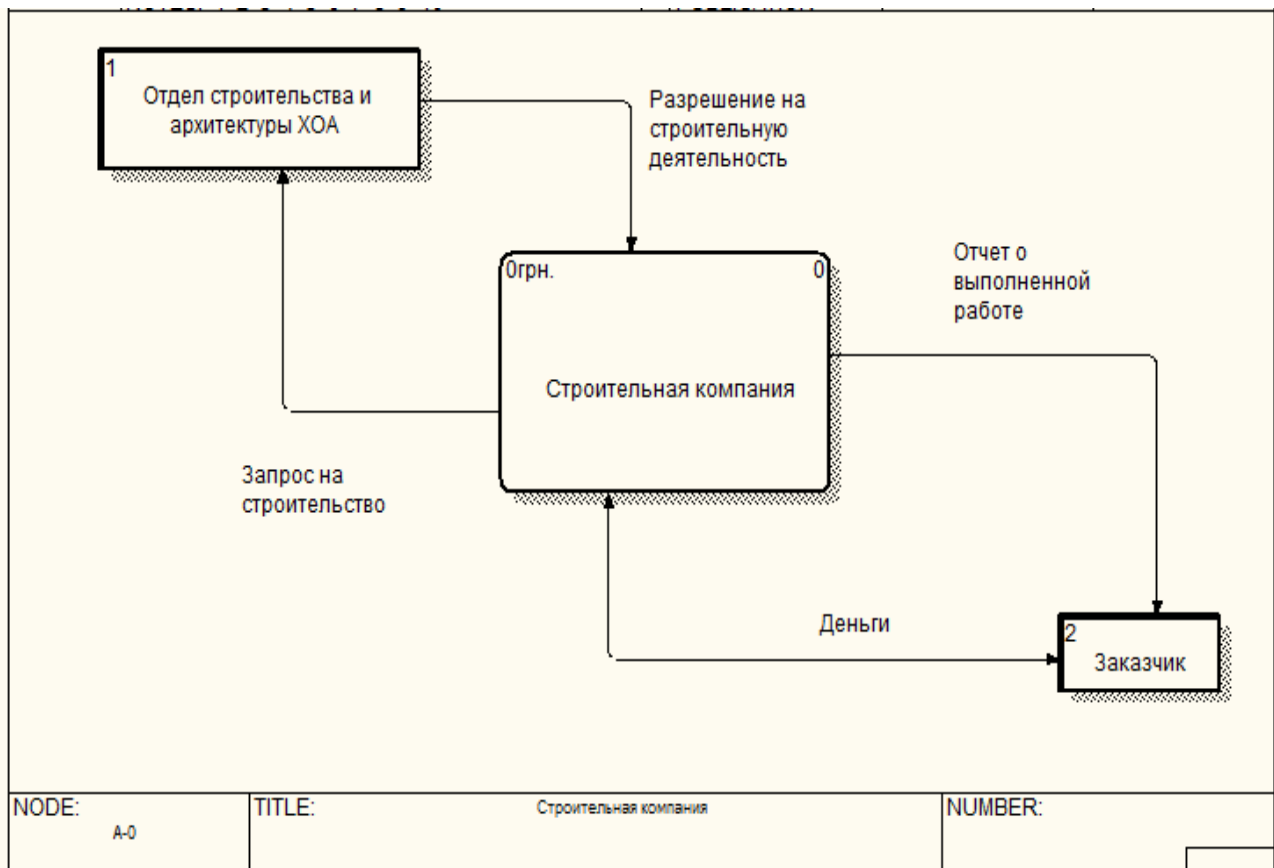


Рисунок 39 – Контекстна діаграма системи «Будівельна компанія»

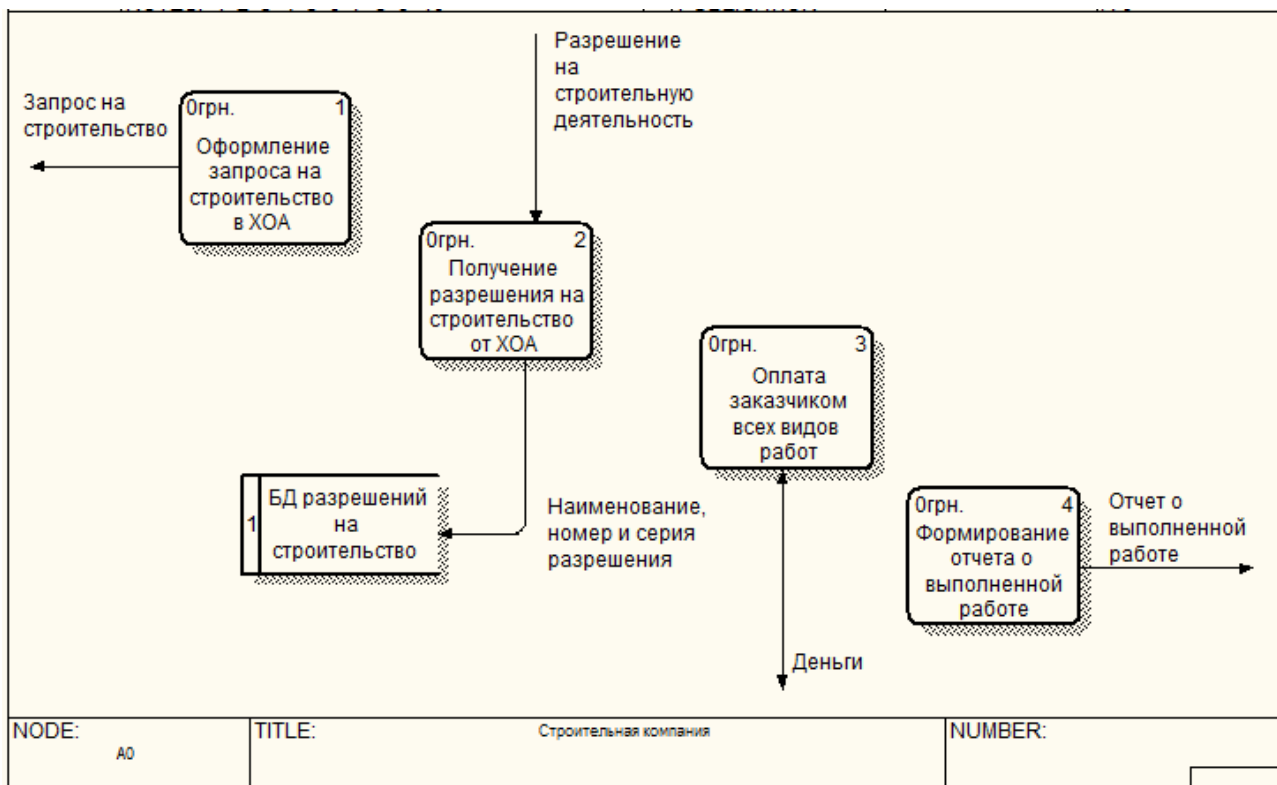


Рисунок 40 – Діаграма декомпозиції 1 рівня системи «Будівельна компанія»

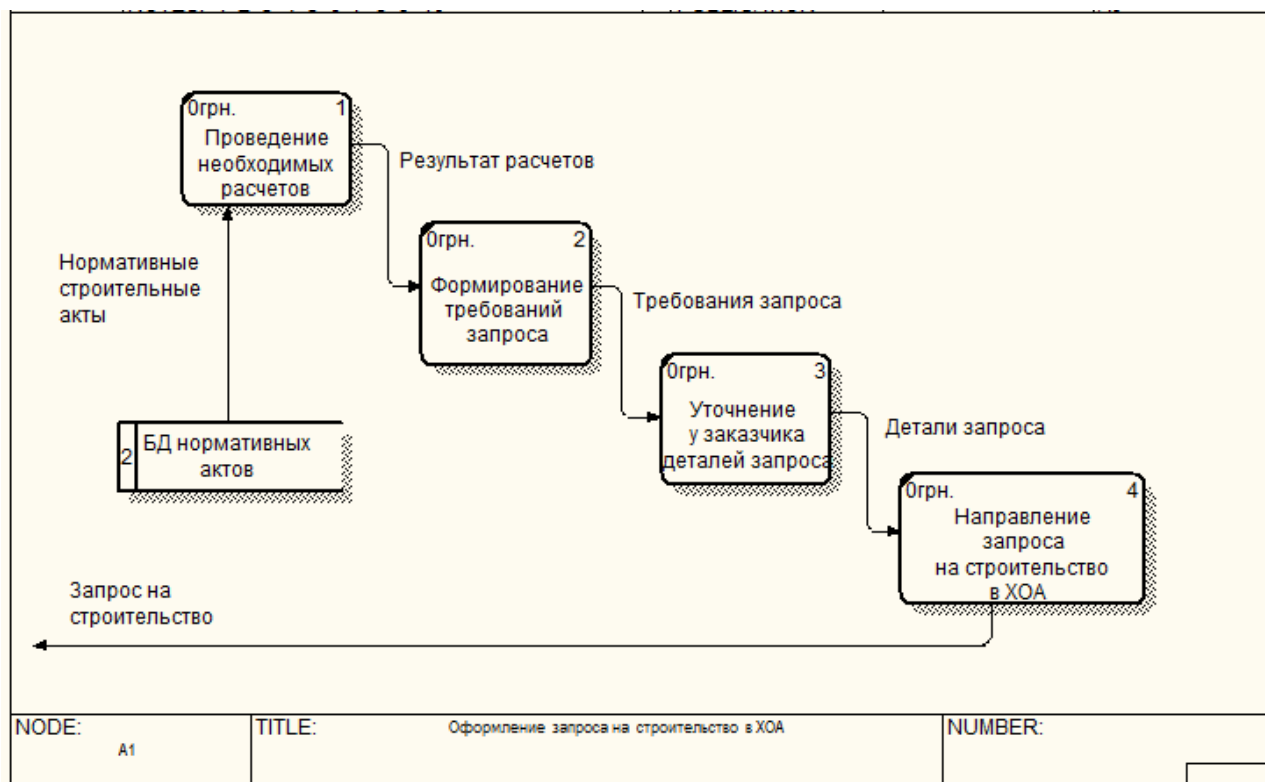


Рисунок 41 – Діаграма декомпозиції 2 рівня процесу
«Оформлення запиту на будівництво в ХОА»



Рисунок 42 – Діаграма декомпозиції 2 рівня процесу
«Отримання дозволу на будівництво від ХОА»

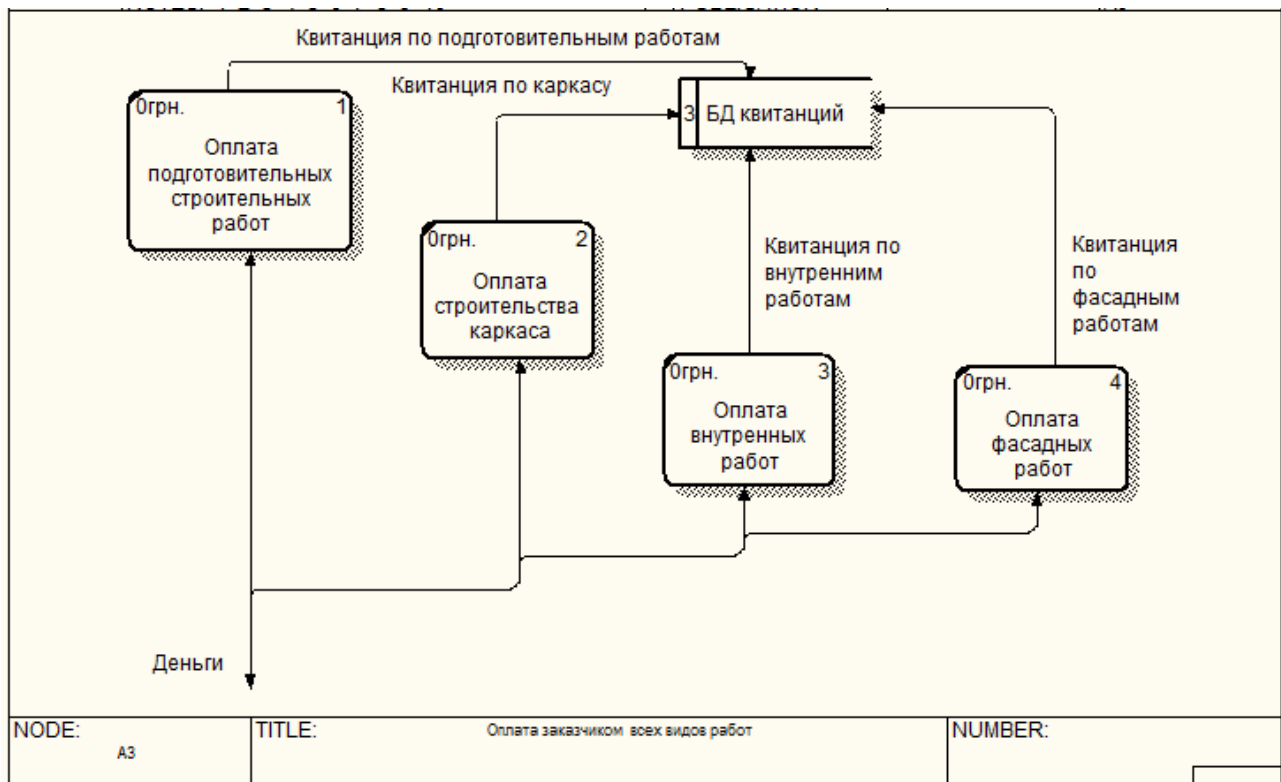


Рисунок 43 – Діаграма декомпозиції 2 рівня процесу
«Оплата заказчиком всех видов работ»

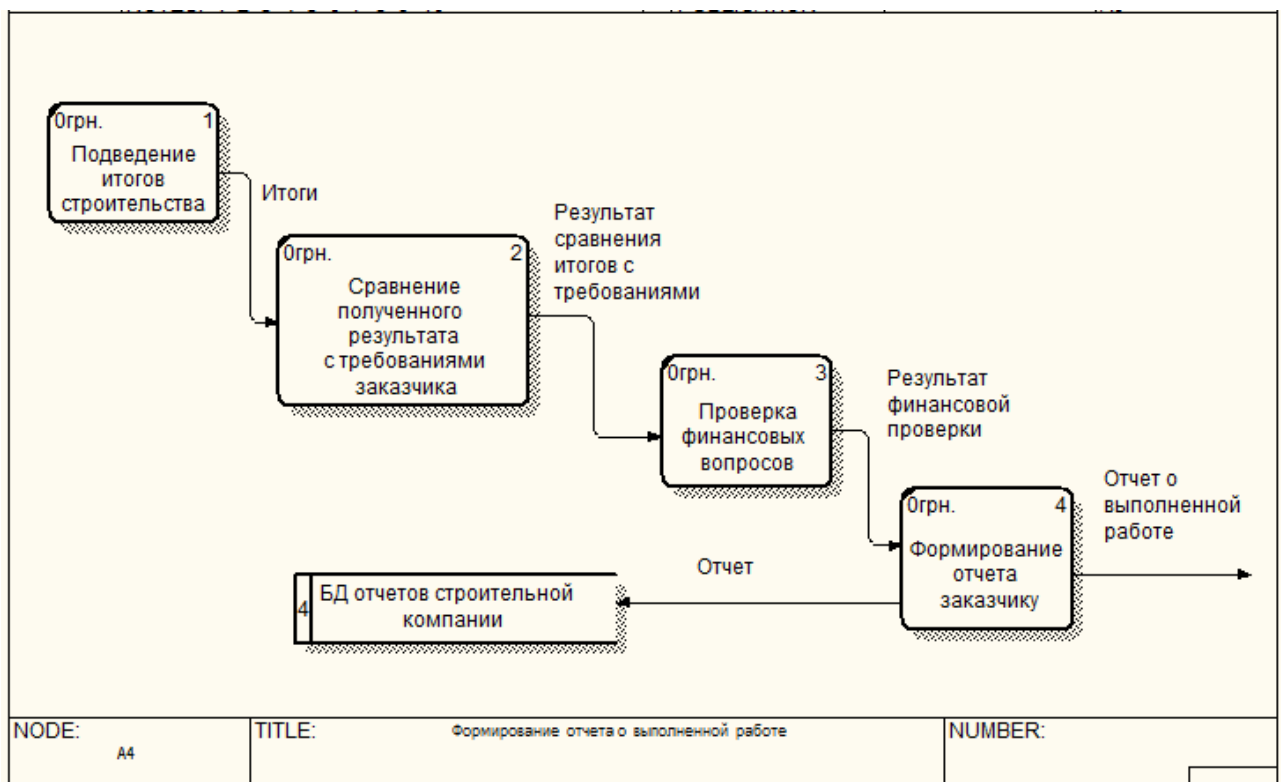


Рисунок 44 – Діаграма декомпозиції 2 рівня процесу
«Формування звіту про виконану роботу»

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Збереження отриманої діаграми.

Збережіть отриману діаграму:

1. У меню **File** виберіть **Save as**.
2. Вкажіть шлях до своєї папки і ім'я файлу **Lab4.bp1**.
3. Натисніть **OK**.

Вказівки з оформлення звіту:

1. Звіт оформлюється у текстовому редакторі Microsoft Word.
2. Звіт повинен містити:
 - назву лабораторної роботи;
 - мета лабораторної роботи;
 - формулювання завдань;
 - короткий опис ходу виконання завдань;
 - діаграми, що відображають результати виконання завдань;
 - висновки по роботі.

Контрольні питання

1. Для чого служить DFD-діаграма?
2. У чому відмінність DFD-діаграми від IDFE0?
3. Який інструмент використовується для побудови зовнішніх сутностей?
4. Яким інструментом можна побудувати сховище даних?
5. Які графічні елементи використовуються для позначення на діаграмі роботи, потоків даних, сховищ даних?

Тестове завдання

1. DFD – це:

- а) діаграма об'єктів та процесів;
- б) діаграма потоку даних;
- в) контекстна діаграма;
- г) функціональна діаграма;
- д) діаграма сутність – зв'язок.

2. Які об'єкти описуються на діаграмі потоків даних:

- а) функції (роботи);
- б) сховища даних;
- в) зовнішні об'єкти;
- г) зовнішні посилання;
- д) всі відповіді правильні.

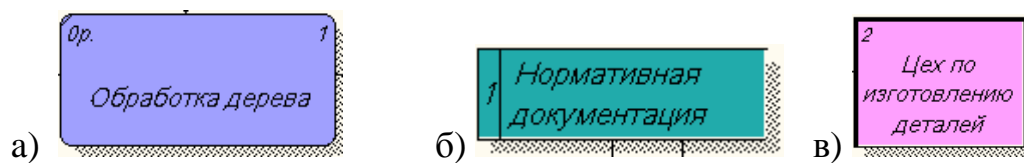
3. Що показують діаграми потоків даних?

- а) зовнішні джерела та приймачі даних;
- б) потоки даних;
- в) сховища (накопичувачі) даних;
- г) всі відповіді правильні;
- д) правильної відповіді немає.






4. Який інструмент потрібно вибрати для створення діаграми DFD?

- а) ; б) ; в) ; г) ; д) .

5. Вкажіть номер рисунка, що позначає на діаграмі сховище даних.



6. Яким інструментом можна позначити в діаграмі DFD зовнішню сутність?

- а) ; б) ; в) ; г) ; д) .

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

1.3 Використання інструментальних засобів для опису логіки взаємодії компонентів об'єктів та процесів

Мета роботи: ознайомлення з функціями інструментального програмного засобу моделювання об'єктів та процесів ERwin, придбання навичок використання методології IDEF3 опису взаємовідносин між процесами обробки інформації та об'єктів, що є частиною цих процесів.

Методичні вказівки по організації роботи студентів

Для опису логіки взаємодії інформаційних потоків більш підходить IDEF3 (**Workflow diagramming**) – методологія моделювання, що використовує графічний опис інформаційних потоків, взаємовідносини між процесами обробки інформації та об'єктів, що є частиною цих процесів.

Діаграми **Workflow** можуть бути використані у моделюванні об'єктів та процесів для аналізу завершеності процедур обробки інформації. З їхньою допомогою можна описувати сценарії дій співробітників організації, наприклад, послідовність обробки замовлення або події, які необхідно обробити за кінцевий час. Кожен сценарій супроводжується описом процесу і може бути використаний для документування кожної функції.

IDEF3 – це метод, основною метою якого є дати можливість аналітикам описати ситуацію, коли процеси виконуються в певній послідовності, а також описати об'єкти, що беруть участь спільно в одному процесі.

Кожна робота в IDEF3 описує який-небудь сценарій об'єкту або процесу і може бути складовою іншої роботи. Оскільки сценарій описує мету і рамки моделі, важливо, щоб роботи іменувалися віддієслівним іменником, що позначає процес дії, або іменним словосполученням, що містить такий іменник.

Діаграма є основною одиницею опису в IDEF3. Важливо правильно побудувати діаграми, оскільки вони призначені для читання іншими людьми, а не тільки автором.

Одиниці роботи – **Unit of Work (UOW)**, також звані роботами (**Activity**), є центральними компонентами моделі. У IDEF3 роботи зображуються прямокутниками з прямими кутами (рис. 45), що мають *назву*, виражену віддієслівним іменником, що позначає процес дії, одиночним або у складі словосполучення, і *номер* (ідентифікатор); інший іменник у складі того ж словосполучення, залежний від віддієслівного іменника, зазвичай, відображає основний вихід (результат).

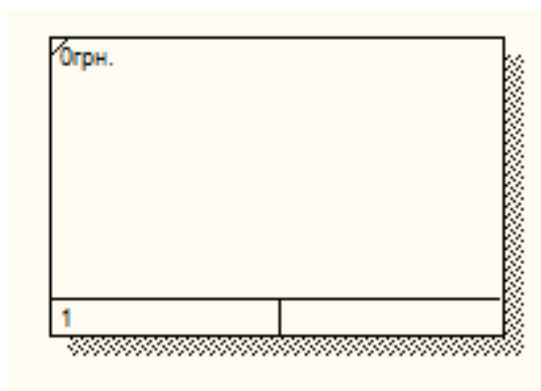


Рисунок 45 – Позначення роботи у діаграмі IDEF3

Зв'язки показують взаємовідносини робіт. Усі зв'язки в IDEF3 односпрямовані і можуть бути спрямовані куди завгодно, але, зазвичай, діаграми IDEF3 намагаються побудувати так, щоб зв'язки були спрямовані зліва направо. В IDEF3 розрізняють три типи стрілок, що зображують зв'язки, стиль яких встановлюється у вкладці **Style** (рис. 46) діалогу **Arrow Properties** (пункт контекстного меню **Style**).

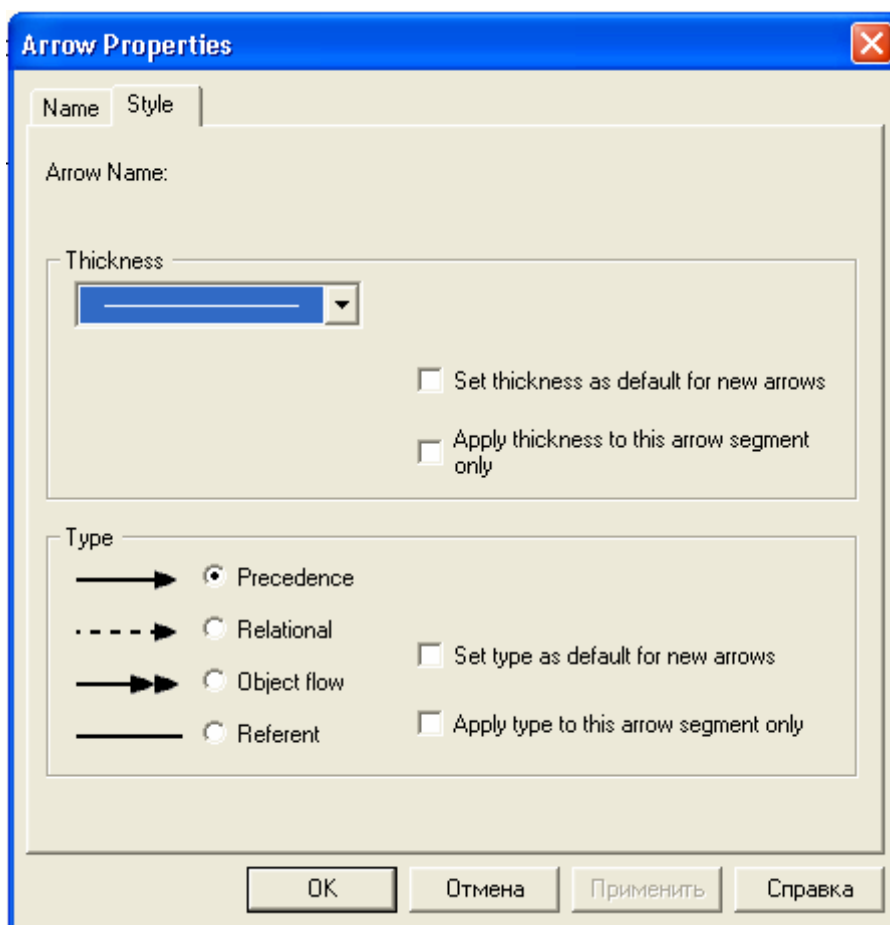
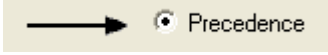
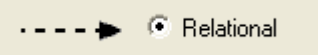



Рисунок 46 – Вкладка **Style** діалогу **Arrow Properties**

Старша (**Precedence**) стрілка  – суцільна лінія, що зв’язує одиниці робіт (**UOW**). Малюється зліва направо або зверху вниз. Показує, що робота-джерело повинна закінчитися перш, ніж робота-мета почнеться.

Стрілка відношень (**Relational**)  – пунктирна лінія, що використовується для зображення зв’язків між одиницями робіт (**UOW**), а також між одиницями робіт і об’єктами посилань.


Потоки об’єктів (**Object Flow**)  – стрілка з двома наконечниками, застосовується для опису того факту, що об’єкт використовується в двох або більше одиницях роботи, наприклад, коли об’єкт породжується в одній роботі і використовується в іншій.

Старший зв’язок показує, що робота-джерело закінчується раніше, ніж починається робота-мета. Часто результатом роботи-джерела стає об’єкт, необхідний для запуску роботи-мети. У цьому випадку стрілку, що позначає об’єкт, зображують з подвійним наконечником. Ім’я стрілки повинне ясно ідентифікувати відображений об’єкт. Потік об’єктів має ту ж семантику, що і старша стрілка.

Відношення показує, що стрілка є альтернативою старшій стрілці або потоку об’єктів у значенні завдання послідовності виконання робіт – робота-джерело не обов’язково повинна закінчитися перш, ніж робота-мета почнеться. Більше того, робота-мета може закінчитися раніше, ніж закінчиться робота-джерело.

Перехрестя (**Junction**). Закінчення однієї роботи може служити сигналом до початку декількох робіт, або ж одна робота для свого запуску може очікувати закінчення декількох робіт. Перехрестя використовуються для відображення логіки взаємодії стрілок при злитті і розгалуженні або для відображення множини подій, які можуть або повинні бути завершені перед початком наступної роботи.

Розрізняють перехрестя для злиття (**Fan-in Junction**) і розгалуження (**Fan-out Junction**) стрілок. Перехрестя не може використовуватися одночасно для злиття і розгалуження.

Для внесення перехрестя служить кнопка  у палітрі інструментів. У діалозі **Select Junction Type** потрібно буде вказати тип перехрестя (рис. 47).

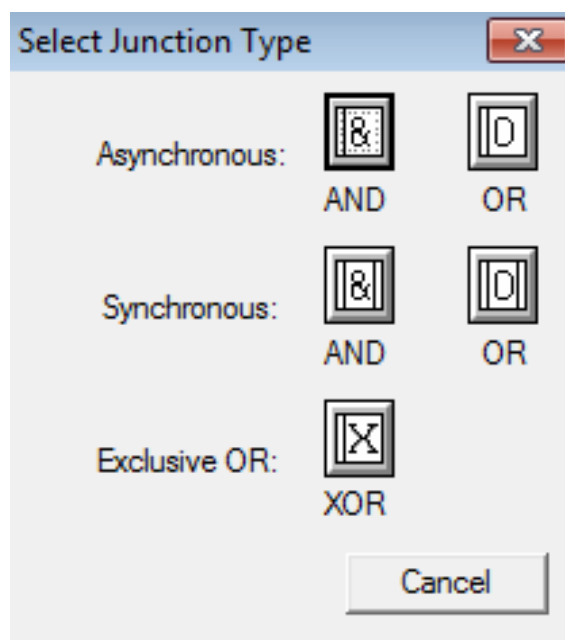


Рисунок 47 – Типи перехресть

Значення кожного типу наведено в таблиці 5.

Таблиця 5 – Типи перехресть

Позначення	Найменування	Значення у разі злиття стрілок Fan-in Junction	Значення у випадку розгалуження стрілок Fan-in Junction
Asynchronous: AND	Асинхронне «І» (Asynchronous AND)	Усі попередні процеси повинні бути завершені	Усі наступні процеси повинні бути запуснені
Synchronous: AND	Синхронне «І» (Synchronous AND)	Усі попередні процеси завершені одночасно	Усі наступні процеси запускаються одночасно
Asynchronous: OR	Асинхронне «АБО» (Asynchronous OR)	Один або кілька попередніх процесів повинні бути завершені	Один або кілька наступних процесів повинні бути запуснені
Synchronous: OR	Синхронне «АБО» (Synchronous OR)	Один або кілька попередніх процесів завершені одночасно	Один або кілька наступних процесів запускаються одночасно
Exclusive OR: XOR	Виключне «АБО» XOR (Exclusive OR)	Тільки один попередній процес завершений	Тільки один наступний процес запускається

Усі перехрестя на діаграмі нумеруються, кожен номер має префікс **J**, як показано на рисунку 48.

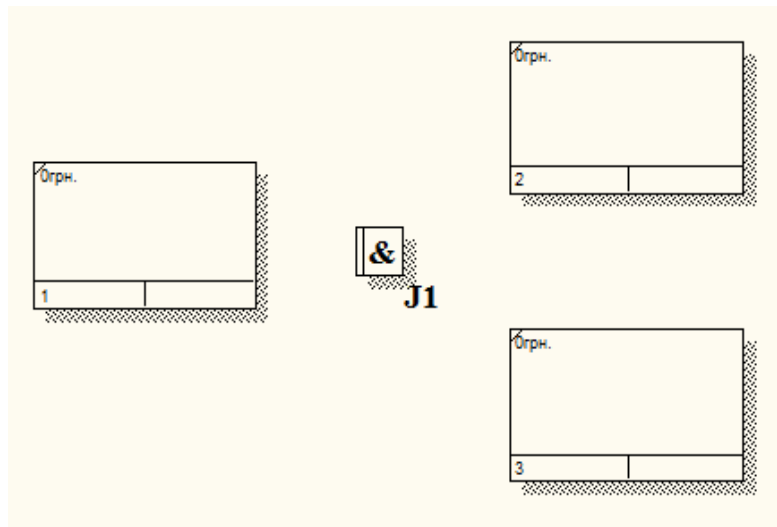


Рисунок 48 – Позначення нумерації перехрестя

Можна редагувати властивості перехрестя (рис. 49) за допомогою діалогу **Junction Properties**, який викликається з контекстного меню.

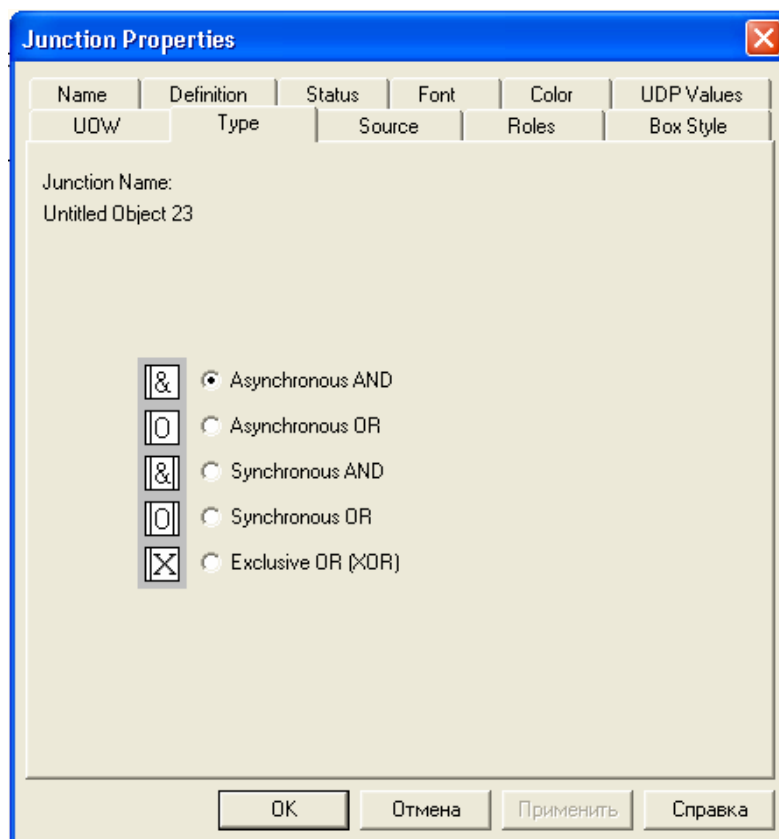


Рисунок 49 – Діалогове вікно властивостей перехресть

На відміну від IDEF0 і DFD в IDEF3 стрілки можуть зливатися і розгалужуватися тільки через перехрестя.

Правила створення перехресть. На одній діаграмі IDEF3 може бути створено кілька перехресть різних типів. Певні поєднання перехресть для злиття і розгалуження можуть призводити до логічних невідповідностей. Щоб уникнути конфліктів, необхідно дотримуватися таких правил:

1. Кожному перехрестю для злиття повинне передувати перехрестя для розгалуження.

2. Перехрестя для злиття типу «І» не може знаходитися за перехрестям для розгалуження типу синхронного або асинхронного «АБО». Дійсно, після роботи № 1 може запускатися тільки одна робота – робота № 2 **або** робота № 3, а для запуску роботи № 4 вимагається закінчення обох робіт – роботи № 2 **і** роботи № 3. Такий сценарій не може реалізуватися (рис. 50).

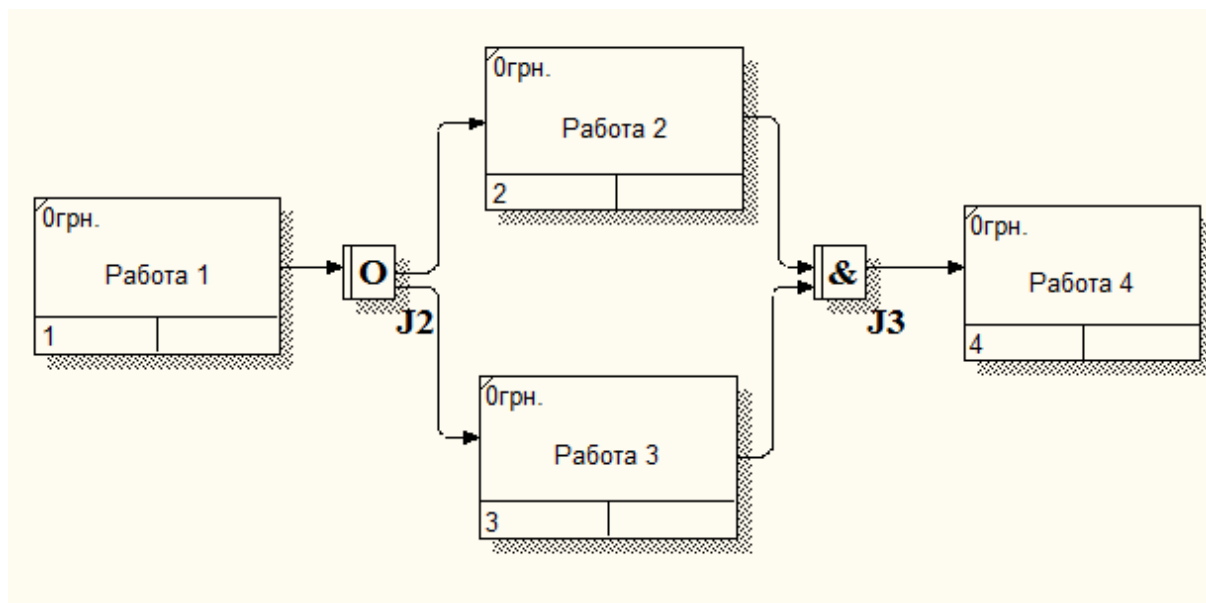


Рисунок 50 – Невірне розміщення перехресть
(перехрестя для злиття типу «І» не може знаходитися за перехрестям для розгалуження типу «АБО»)

3. Перехрестя для злиття типу «І» не може знаходитися за перехрестям для розгалуження типу виключного «АБО» (рис. 51).

4. Перехрестя для злиття типу виключного «АБО» не може знаходитися за перехрестям для розгалуження типу «І» (рис. 52). Тут після завершення роботи № 1 запускаються обидві роботи – робота № 2 **і** робота № 3, а для запуску роботи № 4 потрібно, щоб завершилася **одна і тільки одна** робота – або робота № 2, або робота № 3.

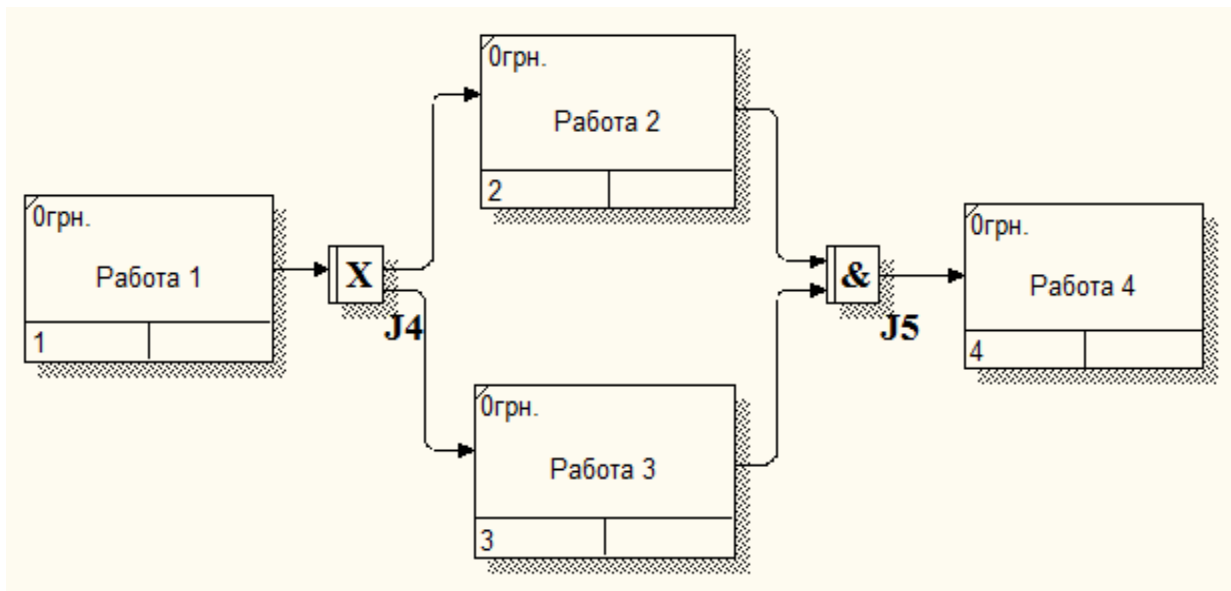


Рисунок 51 – Невірне розміщення перехресть
(перехрестя для злиття типу «I» не може знаходитися за перехрестям для розгалуження типу виключного «АБО»)

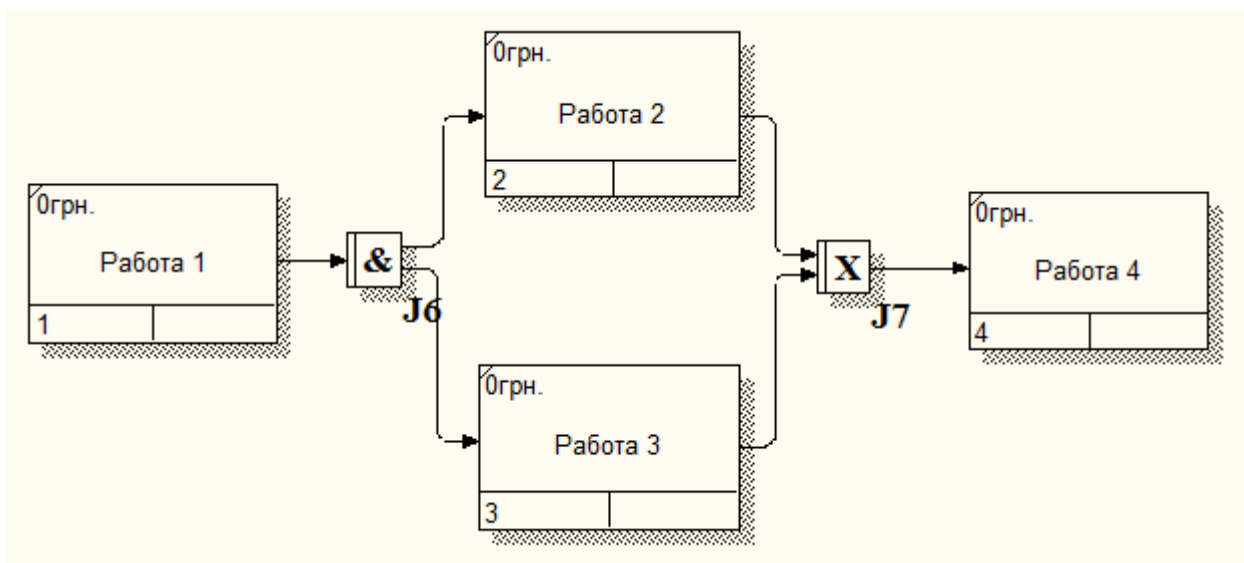


Рисунок 52 – Невірне розміщення перехресть
(перехрестя для злиття типу виключного «АБО» не може знаходитися за перехрестям для розгалуження «I»)

5. Перехрестя, яке має одну стрілку на одній стороні, повинне мати більше однієї стрілки на іншій.

Завдання. Придумати і реалізувати в інструментальному програмному засобі моделювання ERwin свій приклад об'єкту або процесу в методології IDEF3.

Обмеження. Контекстна діаграма повинна містити вхід і вихід. Декомпозиція контекстної діаграми повинна містити не менше 4-х дій. Кожна дія в діаграмі декомпозиції контекстної діаграми має бути декомпованою на не менше, ніж чотири дії. Діаграми декомпозицій повинні містити: всі типи стрілок (**Precedence, Relational, Object flow**), перехрестя злиття (**Fan-in junction**) і розгалуження (**Fan-out junction**), всі типи перехресть (**Asynchronous AND, Synchronous AND, Asynchronous OR, Synchronous OR , Exclusive OR**) і об'єкт посилення (**Referent**).

Створення діаграми IDEF3.

1. Відкрийте файл **Lab4.bp1**, збережений на попередньому уроці.
2. Виберіть в діалозі **CA ERwin Process Modeler** радіо-кнопку **Process Flow** (IDEF3) (рис. 53).
3. Внесіть назву моделі «Побудова будинку» і клацніть по кнопці **OK**.

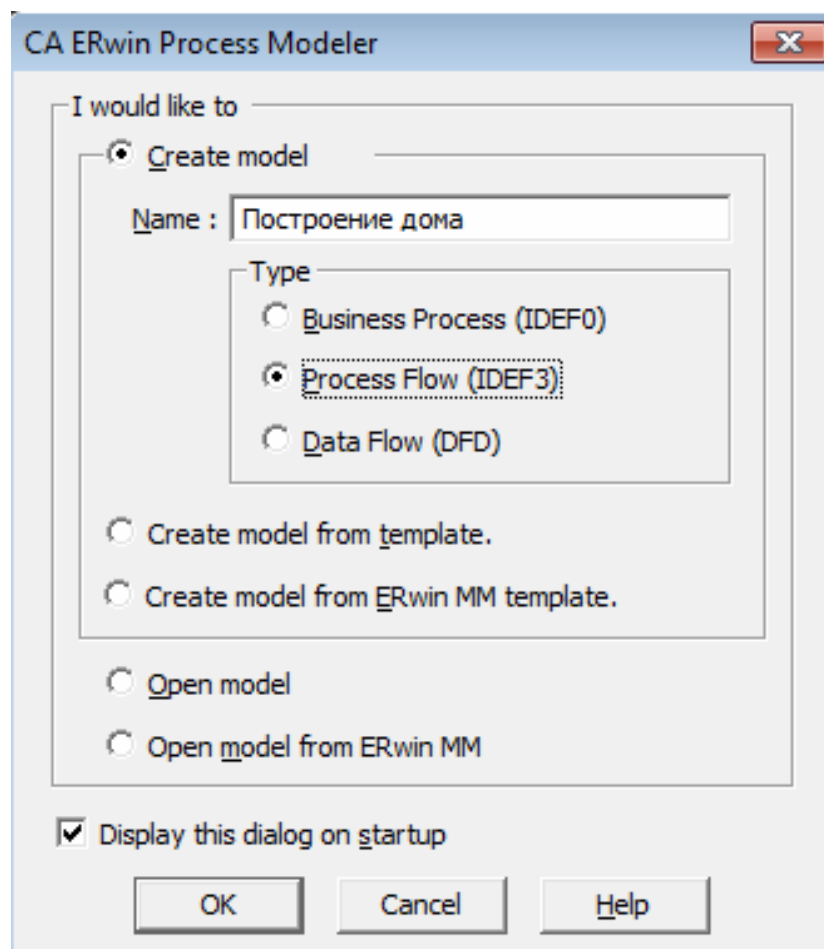


Рисунок 53 – Діалог **CA ERwin Process Modeler**

4. Правою кнопкою миші клацніть по роботі, виберіть у контекстному меню **Name** і внесіть ім'я роботи «Побудова будинку».

5. У вкладці **Definition** внесіть визначення «Описываются все процессы построения дома» (рис. 54).

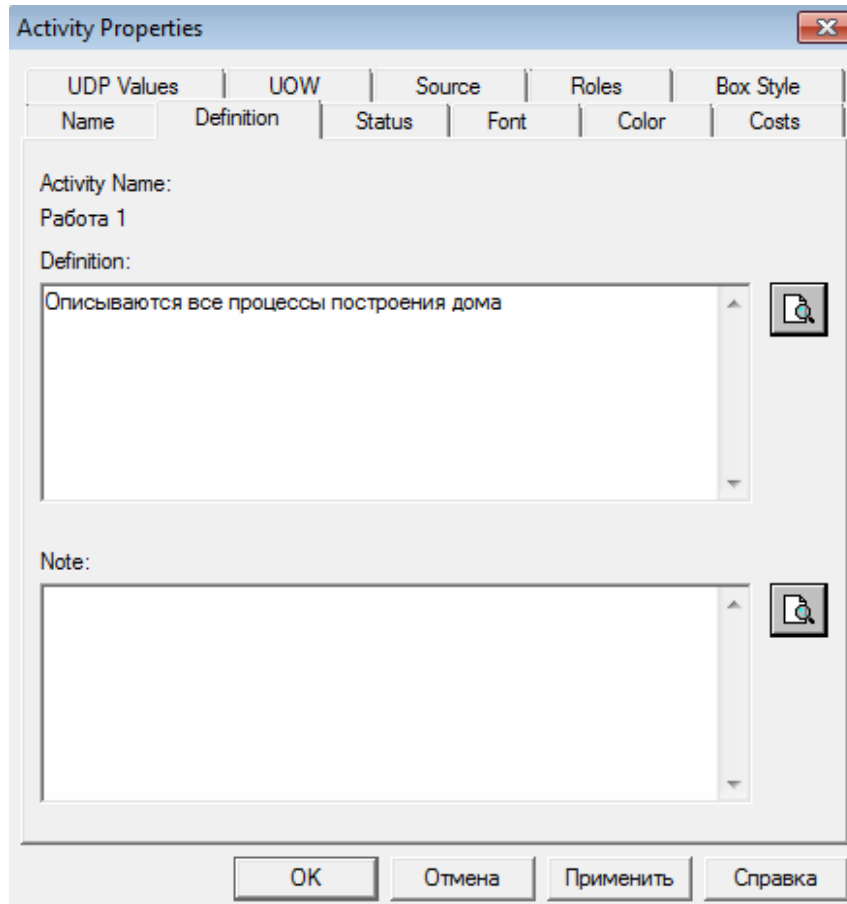


Рисунок 54 – Діалогове вікно властивостей роботи

5. Створіть діаграму декомпозиції (рис. 55).

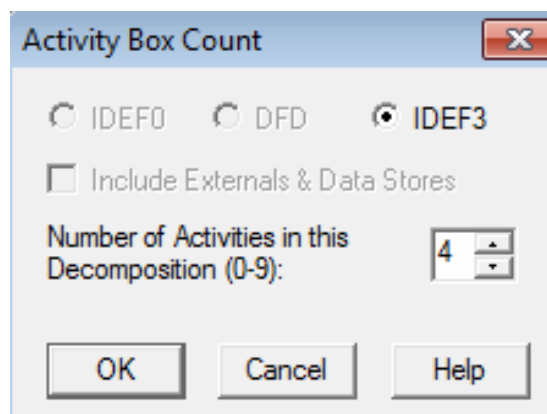



Рисунок 55 – Діалог **Activity Box Count**

6. Внесіть додаткові роботи (кнопка ).

7. Внесіть імена робіт «Підготовка до будівництва будинку», «Зведення будинку», «Проведення внутрішніх робіт», «Проведення фасадних робіт», «Здача будинку».

Додавання в діаграму IDEF3 об'єкта посилання.

1. За допомогою кнопки (додати в діаграму об'єкт посилання – **Referent**), розташованої на палітрі інструментів, створіть об'єкт посилання.

2. Внесіть ім'я об'єкта зовнішнього посилання «Документи».

3. Змініть стиль стрілки на **Referent** (рис. 56).

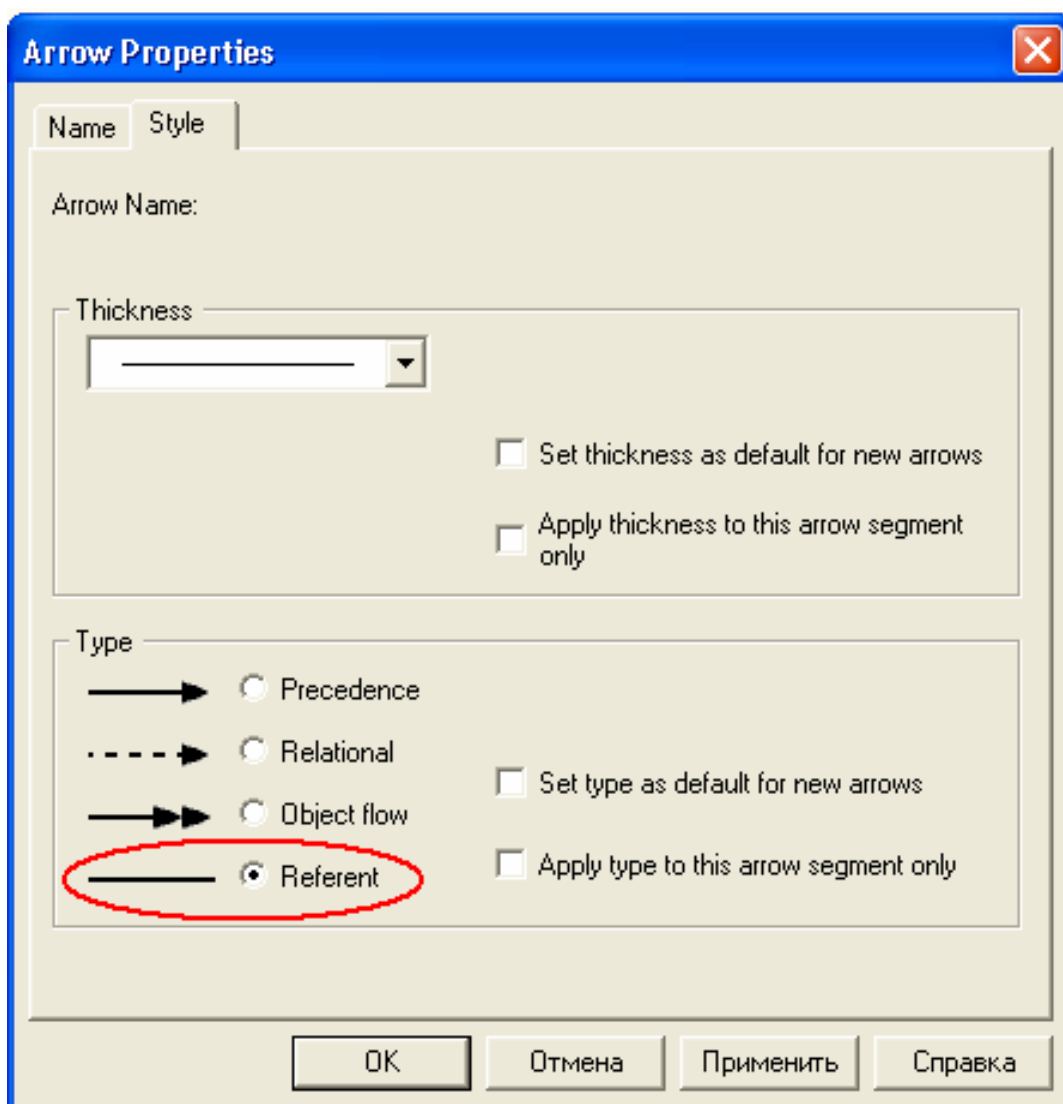


Рисунок 56 – Діалогове вікно визначення стилю стрілок

Об'єкт посилання в IDEF3 виражає якусь ідею, концепцію або дані, які не можна пов'язати зі стрілкою, перехрестям або роботою.

Зв'язування робіт за допомогою стрілок.

1. Зв'яжіть стрілками роботи «Оформлення документів на земельну ділянку» (вихід), «Планування строків робіт» та «Підготовка ділянки до початку роботи».

2. Змініть стиль стрілки на **Object Flow**  **Object flow** (рис. 57).

В IDEF3 ім'я стрілки може бути відсутнім, хоча ERwin показує відсутність імені як помилку.

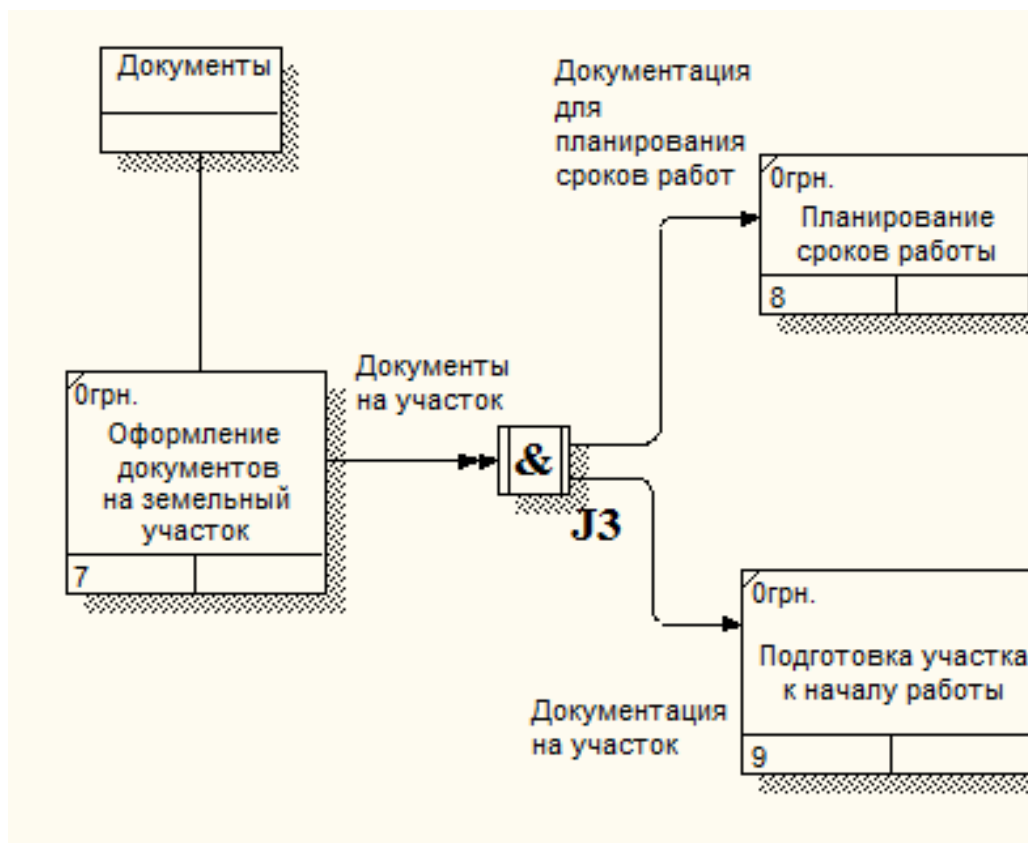




Рисунок 57 – Результат створення робіт та об'єкта посилання

3. Якщо роботи виконуються послідовно, то з'єднайте їх стрілкою **Precedence**  **Precedence**, вона показує, що робота-джерело повинна закінчитися перш, ніж робота-мета почнеться.

Встановлення перехрестя.

1. За допомогою кнопки  на палітрі інструментів внесіть одне перехрестя типу асинхронного «АБО» (один або декілька наступних процесів повинні бути запущені) і зв'яжіть роботи з перехрестям (рис. 58), це випадок розгалуження стрілок **Fan-in Junction**.

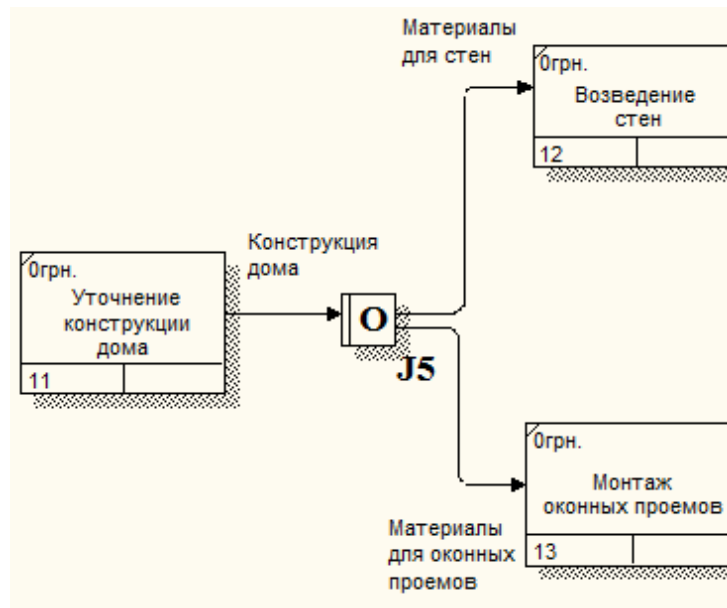


Рисунок 58 – Результат створення перехрестя

2. Правую кнопку клацните по перехрестю для розгалуження, виберіть Name і внесіть ім'я «Материалы, необходимые для стен та вікон».
3. Самостійно встановіть перехрестя для злиття стрілок.
4. Перевірте себе (рис. 59 – рис. 65).

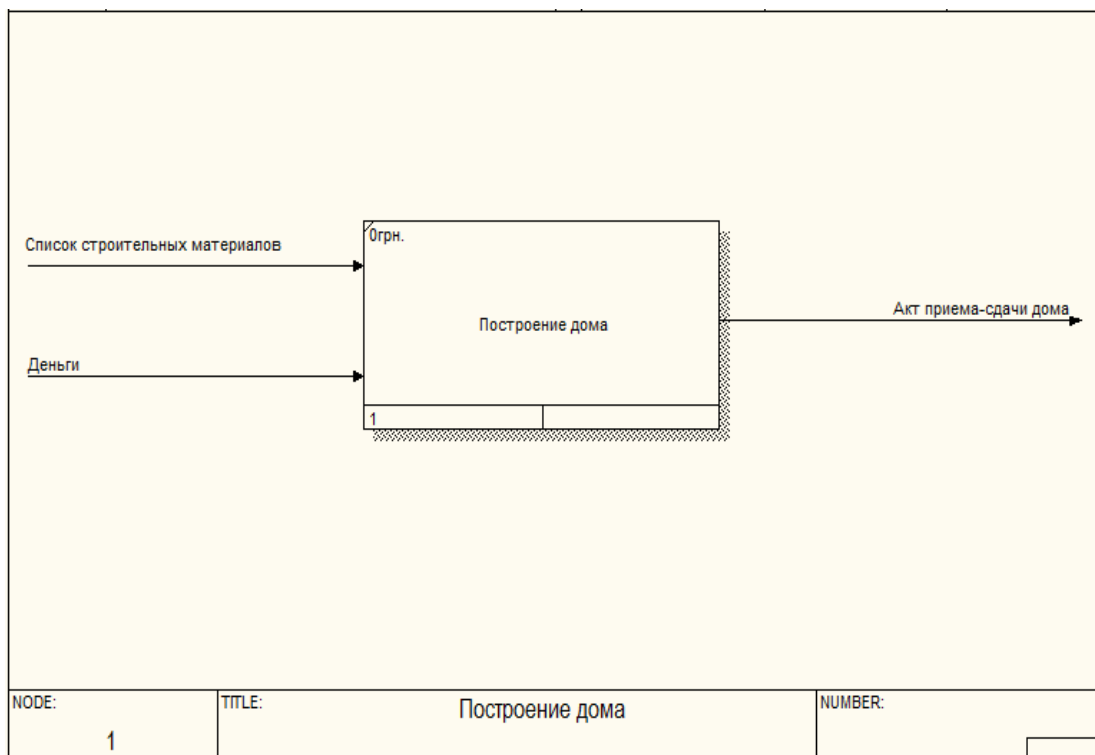


Рисунок 59 – Контекстна діаграма процесу «Побудова будинку» у нотації IDEF3

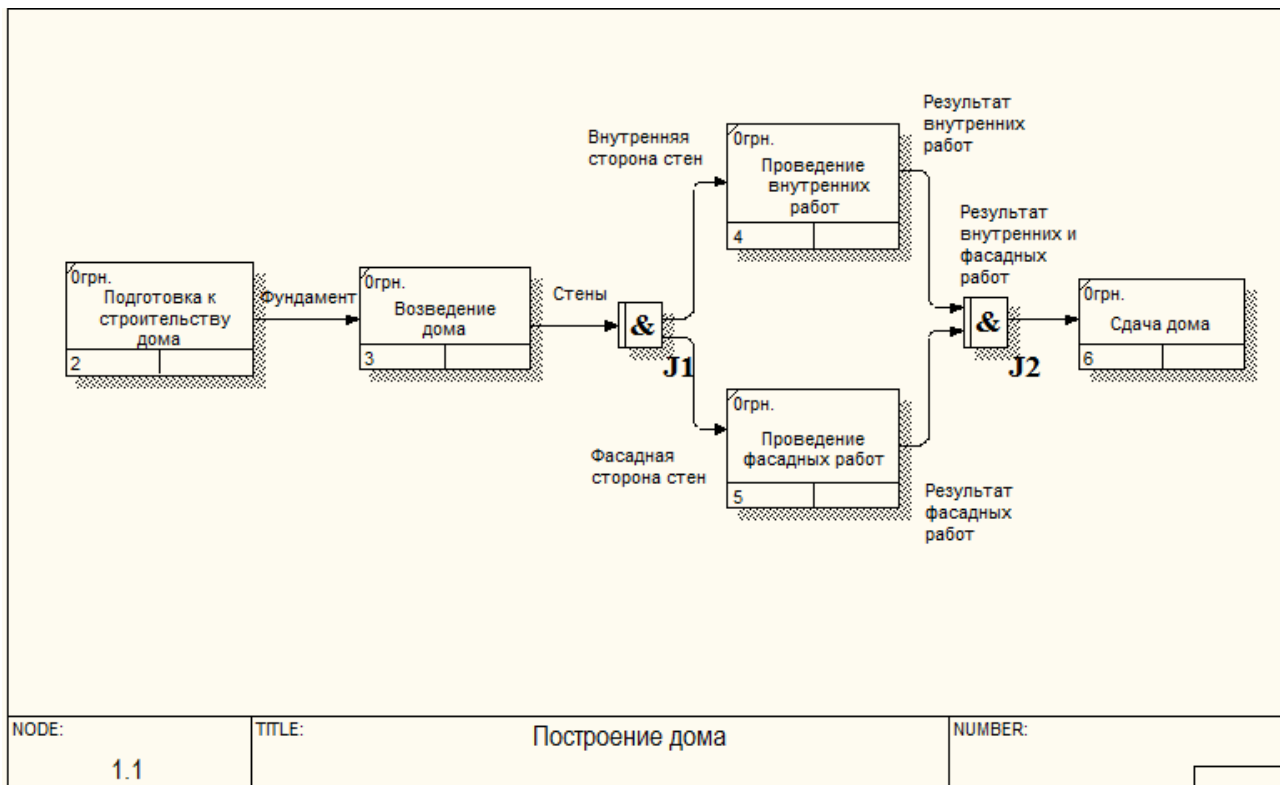


Рисунок 60 – Диаграмма декомпозиции 1 уровня процесса «Построение здания»

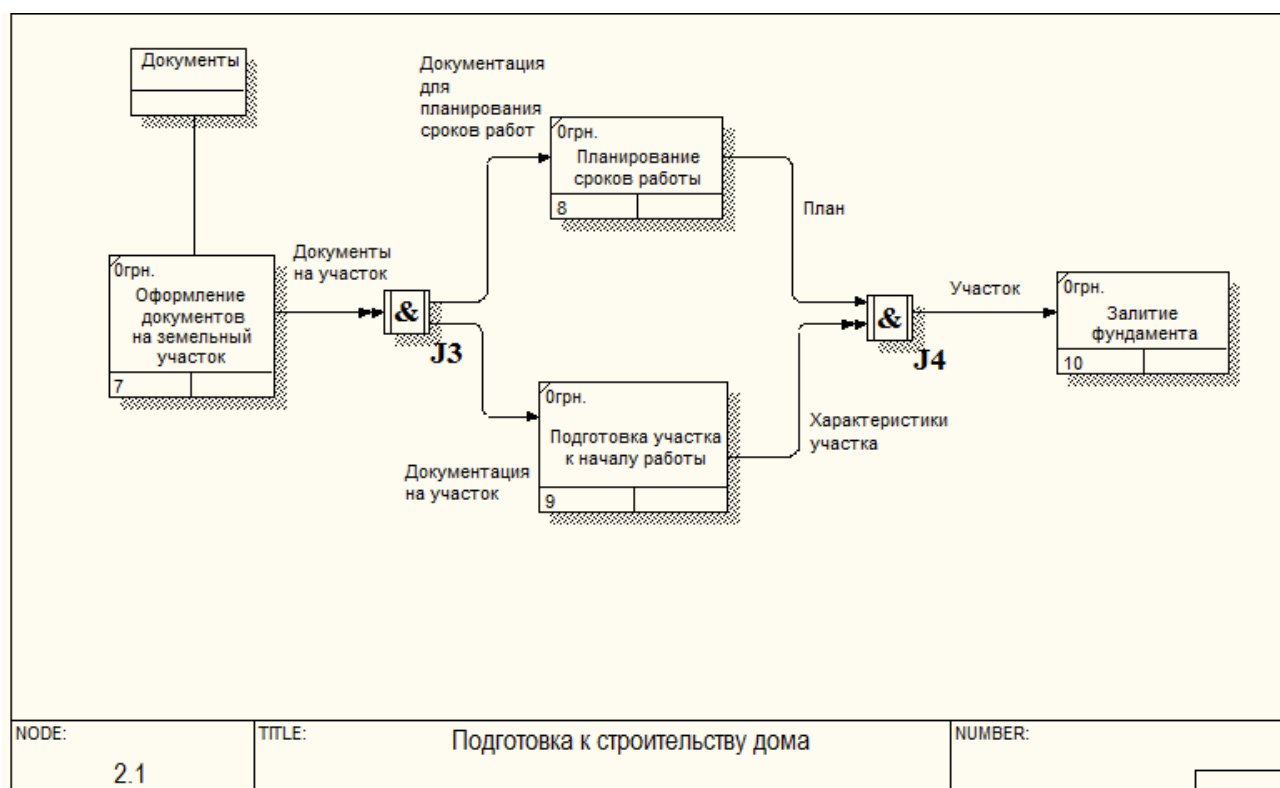


Рисунок 61 – Диаграмма декомпозиции 2 уровня процесса «Подготовка к строительству здания»

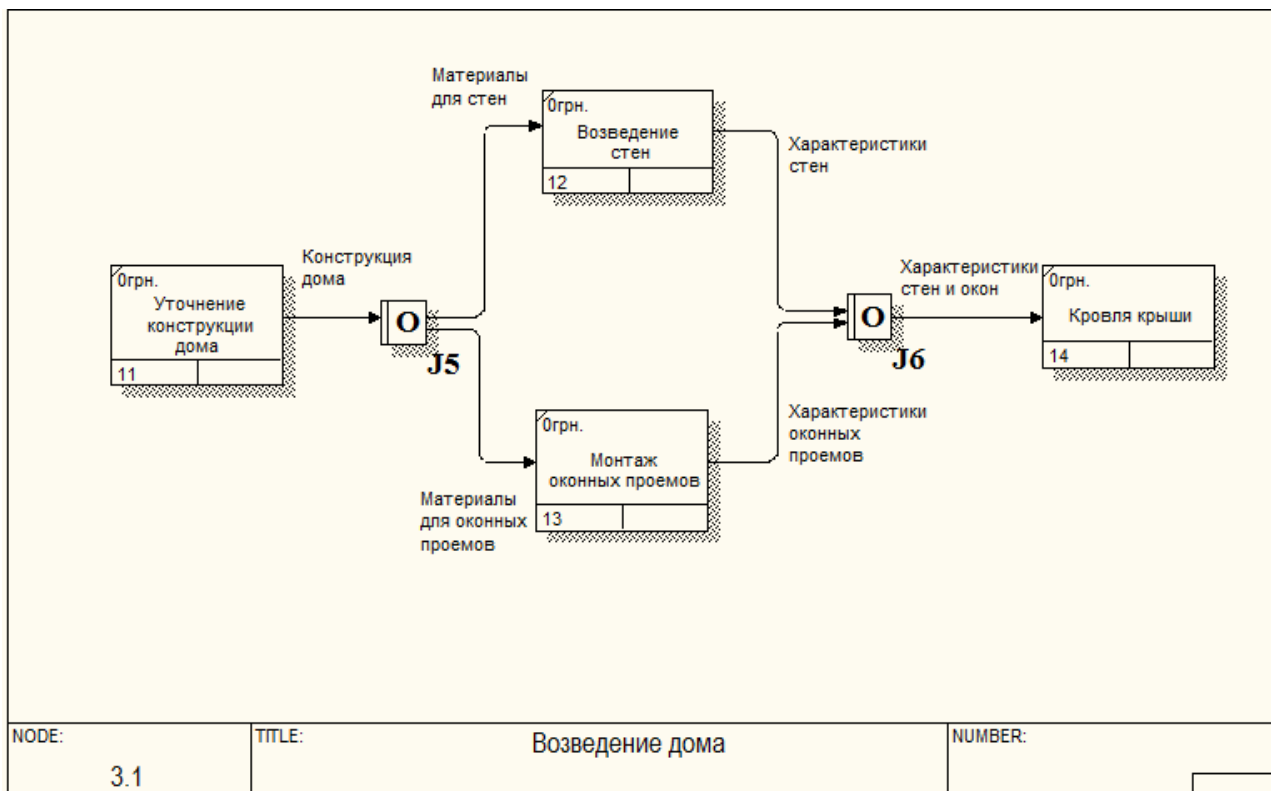


Рисунок 62 – Діаграма декомпозиції 2 рівня процесу «Зведення стін»

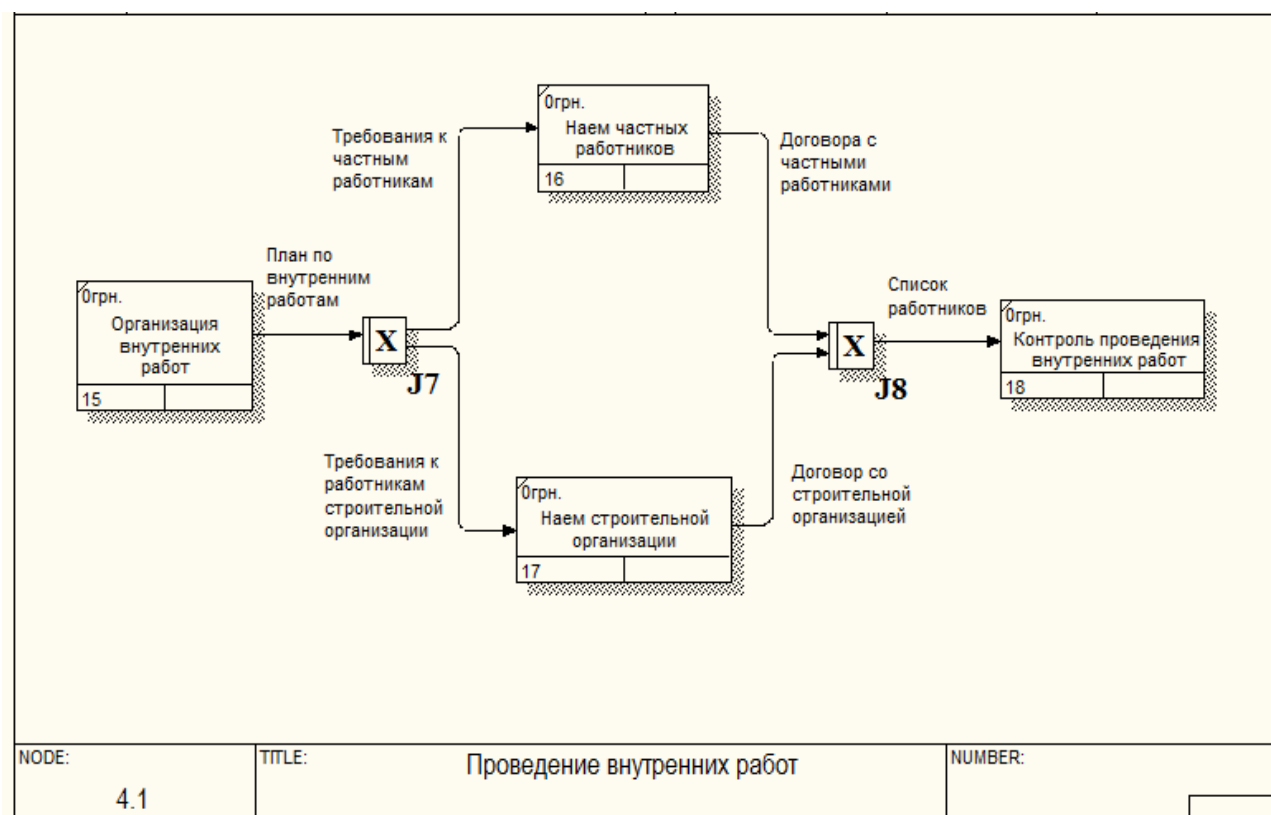


Рисунок 63 – Діаграма декомпозиції 2 рівня процесу «Проведення внутрішніх робіт»

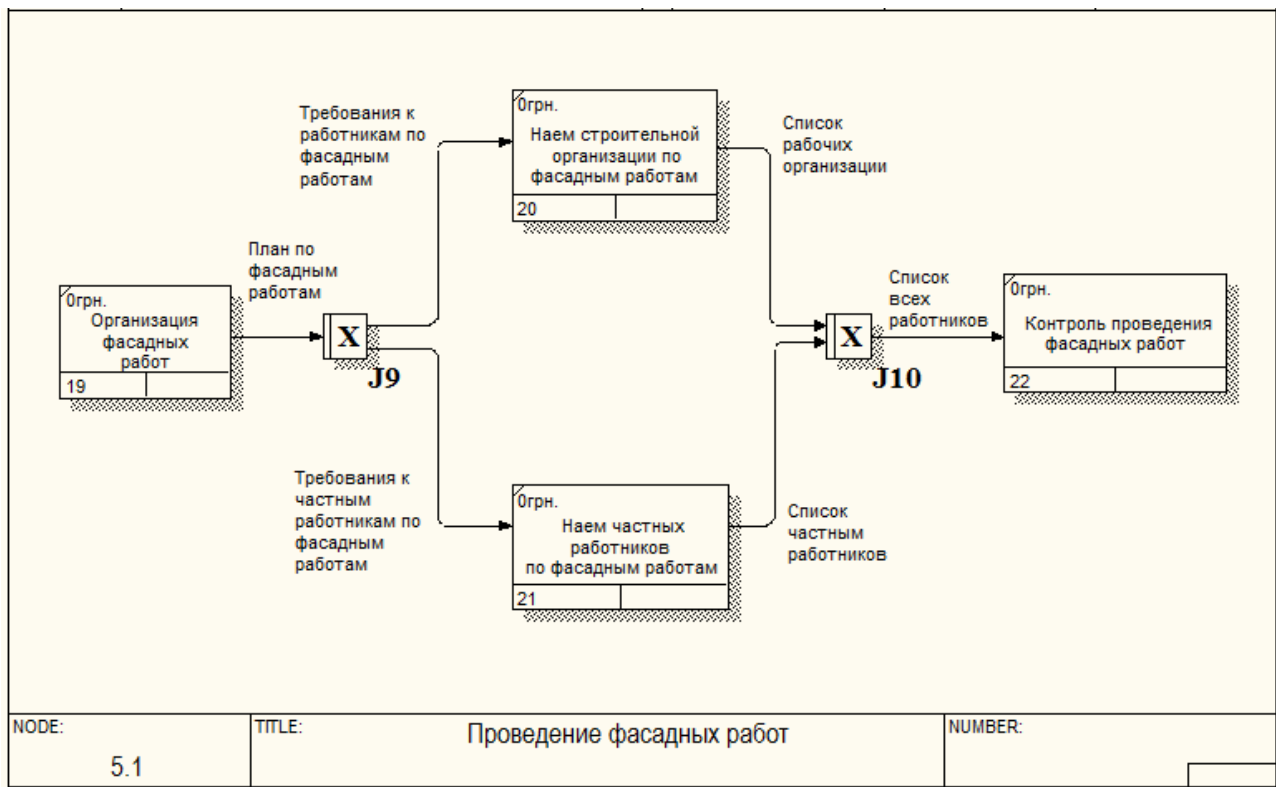


Рисунок 64 – Діаграма декомпозиції 2 рівня процесу
«Проведення фасадних робіт»

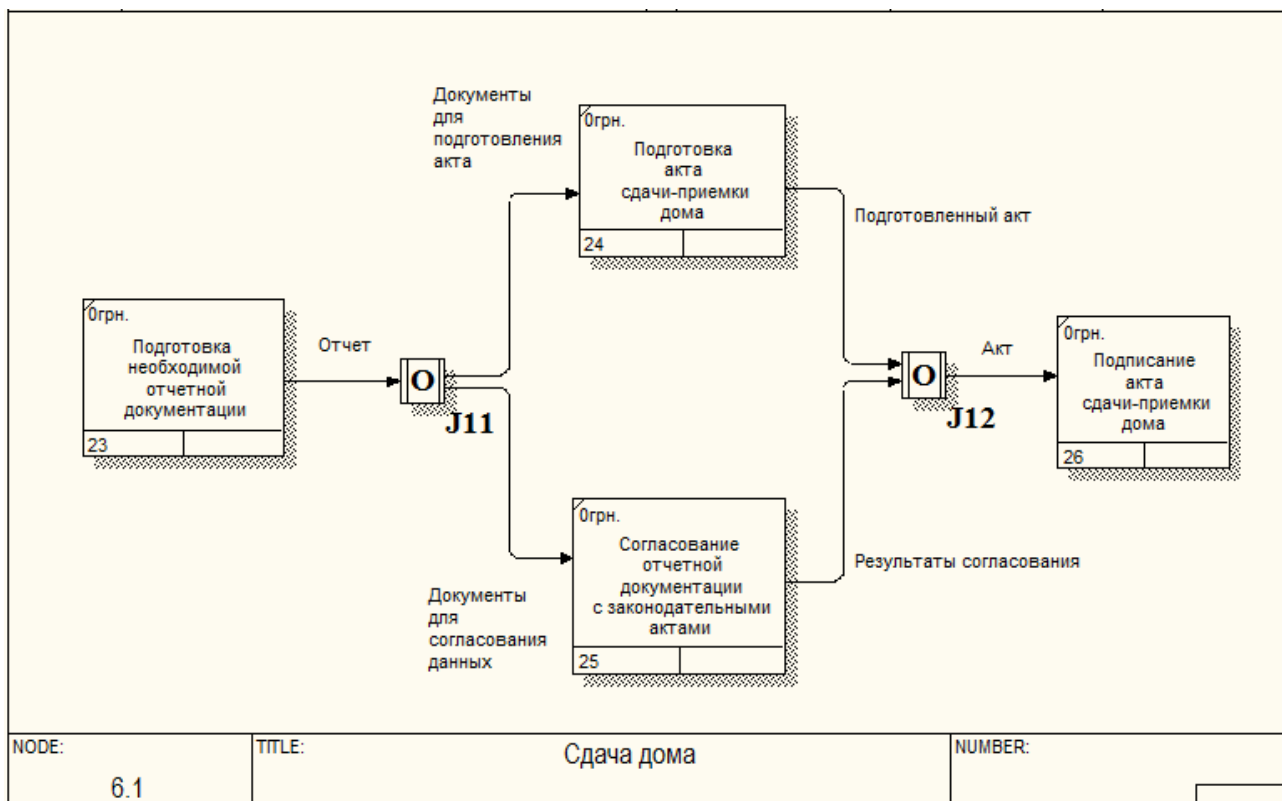
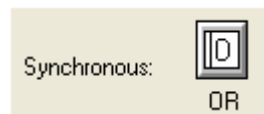


Рисунок 65 – Діаграма декомпозиції 2 рівня процесу «Здача будинку»

Примітка. Так як робота «Підписання акту здачі-прийому будинку» повинна проводитися після закінчення всіх попередніх робіт, то обраний тип перехрестя – синхронне «АБО»



Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Збереження отриманої діаграми.

Збережіть отриману діаграму:

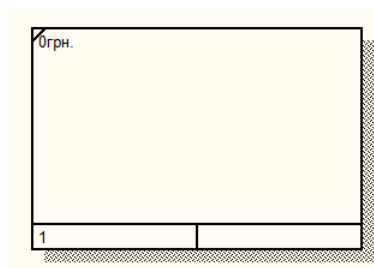
1. У меню **File** виберіть **Save as**.
2. Вкажіть шлях до своєї папки і ім'я файлу **Lab5.bp1**.
3. Натисніть OK.

Контрольні питання

1. Для чого будується діаграма IDEF3?
2. Чим діаграма IDEF3 відрізняється від діаграми IDEF0?
3. Як графічно позначається робота в діаграмі IDEF3?
4. З якою метою між роботами встановлюють перехрестя?
5. Які типи перехресть вам знайомі?

Тестове завдання

1. В якій нотації використовується таке графічне зображення роботи:








- а) DFD;
- б) IDEF3;
- в) IDEF0;
- г) правильні відповіді б), в);
- д) правильні відповіді а), в).

2. Вкажіть невірну назву роботи:






- а) «Обслуговування клієнта»;
- б) «Оформлення замовлення»;
- в) «Якісна фурнітура»;
- г) «Обробка дерева»;
- д) правильної відповіді немає.

3. За допомогою якого інструменту можна встановити між роботами перехрестя?






- а)  ; б)  ; в)  ; г)  ; д) .

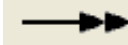
4. Розподіліть перехрестя в таблиці, відповідно до їх змісту.

Позначення	Значення у разі злиття стрілок Fan-in Junction	Значення у випадку розгалуження стрілок Fan-in Junction
	Усі попередні процеси повинні бути завершені	Усі наступні процеси повинні бути запуснені
	Усі попередні процеси завершені одночасно	Усі наступні процеси запускаються одночасно
	Один або кілька попередніх процесів повинні бути завершені	Один або кілька наступних процесів повинні бути запуснені
	Один або кілька попередніх процесів завершені одночасно	Один або кілька наступних процесів запускаються одночасно
	Тільки один попередній процес завершений	Тільки один наступний процес запускається

- а) Exclusive OR:  XOR ; б) Synchronous:  OR ; в) Asynchronous:  OR ;
- г) Asynchronous:  AND ; д) Synchronous:  AND .

5. За допомогою якого інструменту можна додати в діаграму об'єкт-посилання?

- а) ; б) ; в) ; г) ; д) .

6. Що показує стрілка  **Object flow**, що з'єднує роботи?

- а) стрілка показує, що робота-джерело повинна закінчитися перш, ніж робота-мета почнеться;
б) стрілка показує зв'язок між одиницями робіт і об'єктами посилань;
в) стрілка показує, що об'єкт використовується в двох або більше одиницях роботи;
г) всі відповіді правильні.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Вказівки з оформлення звіту:

1. Звіт оформлюється у текстовому редакторі Microsoft Word.
2. Звіт повинен містити:
 - назву лабораторної роботи;
 - мета лабораторної роботи;
 - формулювання завдань;
 - короткий опис ходу виконання завдань;
 - діаграми, що відображають результати виконання завдань;
 - висновки по роботі.

1.4 Використання інструментальних засобів для опису змішаної моделі об'єктів та процесів

Мета роботи: ознайомлення з функціями інструментального засобу ERwin для опису змішаної моделі об'єктів і процесів.

Методичні вказівки по організації роботи студентів

Вивчіть три етапи створення змішаних моделей за методологією IDEF0, DFD та IDEF3, що реалізуються програмним засобом ERwin:

- побудова контекстної діаграми (методологія IDEF0);
- декомпозиція контекстної діаграми (методологія IDEF0);
- діаграми другого рівня (методології IDEF0, DFD та IDEF3).

Завдання. Придумати і реалізувати в інструментальному програмному засобі моделювання ERwin свій приклад опису змішаної моделі об'єктів і процесів.

Обмеження. Контекстна діаграма повинна містити всі типи граничних стрілок (вхід, вихід, управління та механізм), методологія IDEF0 (рис. 66).

Декомпозиція контекстної діаграми повинна містити не менше 4-х дій. Кожна дія в діаграмі декомпозиції контекстної діаграми повинна мати вхідну стрілку, вихідну стрілку, управління та механізм (методологія IDEF0) (рис. 67 – рис. 69).

Одна дія в діаграмі декомпозиції контекстної діаграми має бути декомпозованою за методологією IDEF3. Діаграма декомпозицій повинна містити не менше 4-х дій, перехрестя і об'єкт посилення, з'єднані між собою стрілками (рис. 70).

Одна дія в діаграмі декомпозиції контекстної діаграми має бути декомпозованою за методологією DFD. Діаграма декомпозиції повинна містити не менше 4-х дій і сховище даних, з'єднаних між собою стрілками (рис. 71).

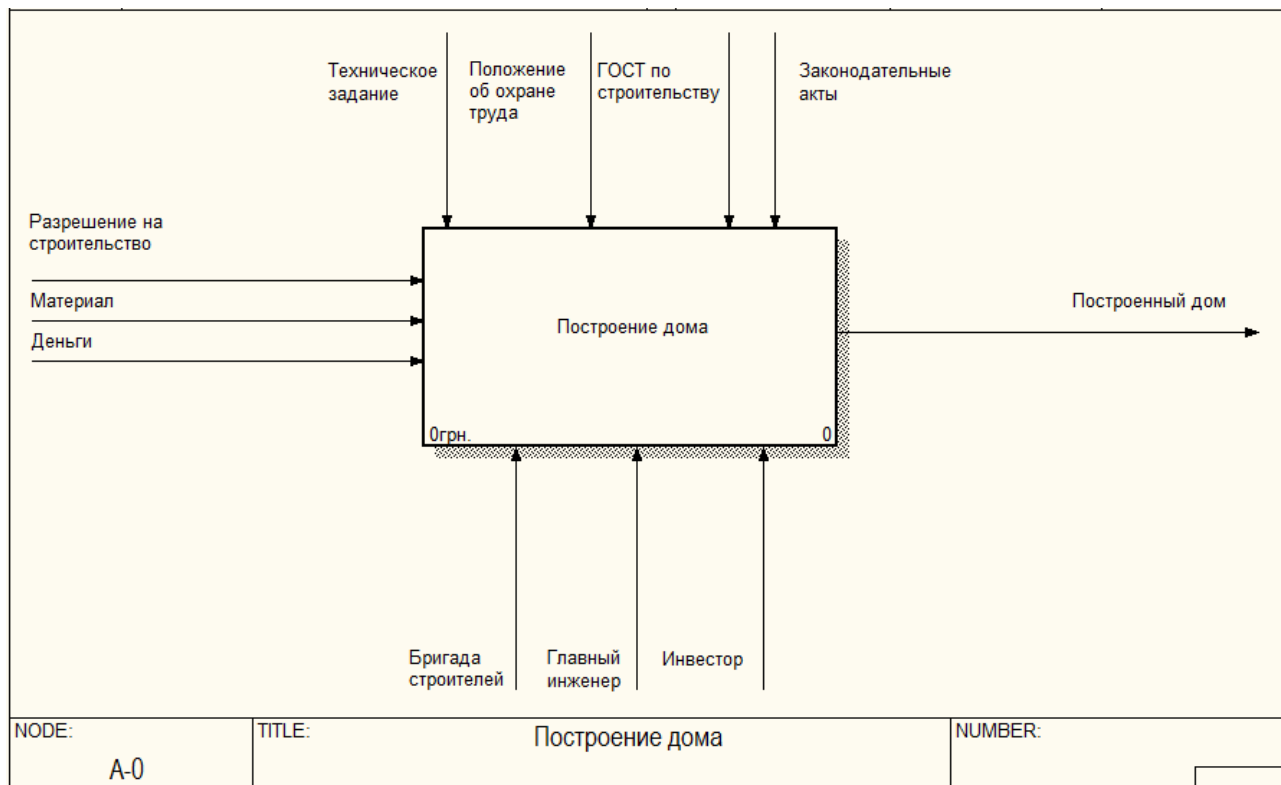


Рисунок 66 – Контекстна діаграма процесу «Побудова будинку»

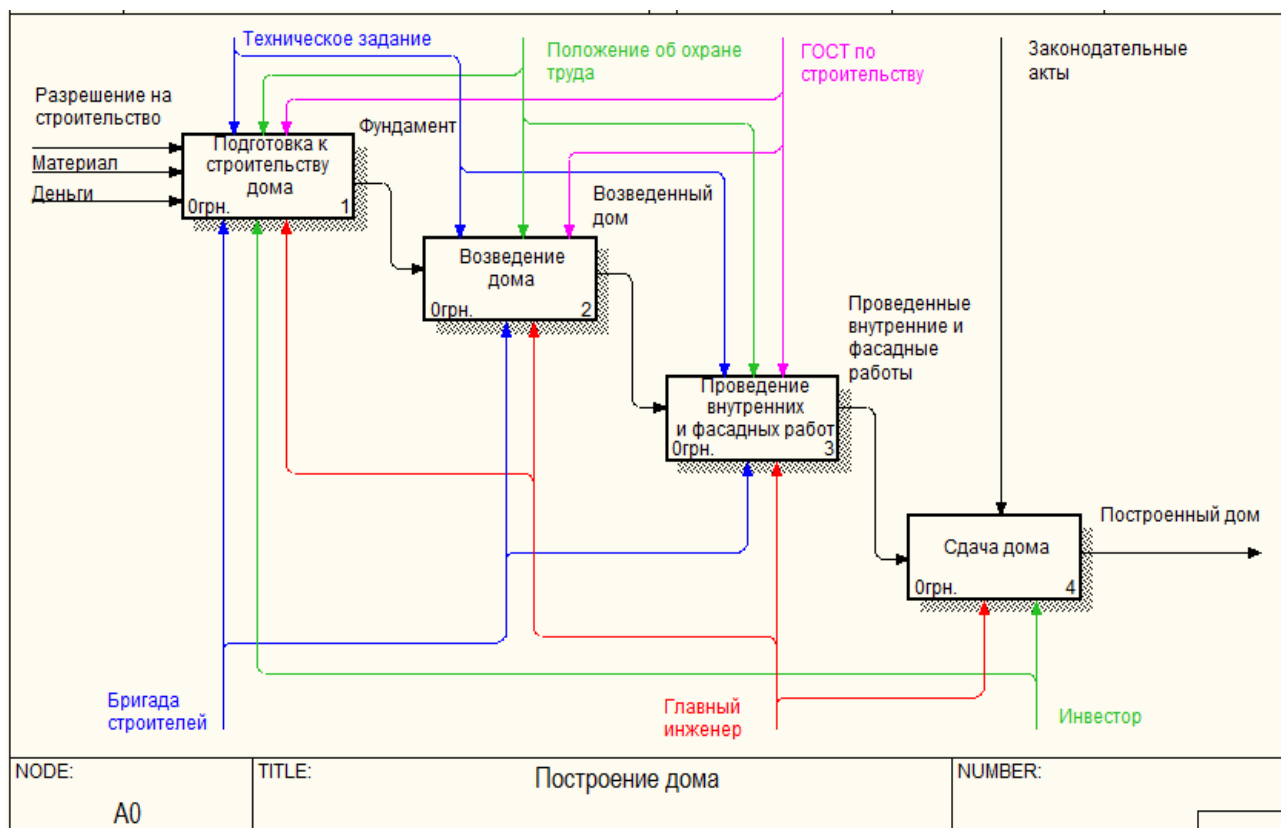


Рисунок 67 – Діаграма декомпозиції першого рівня процесу «Побудова будинку»

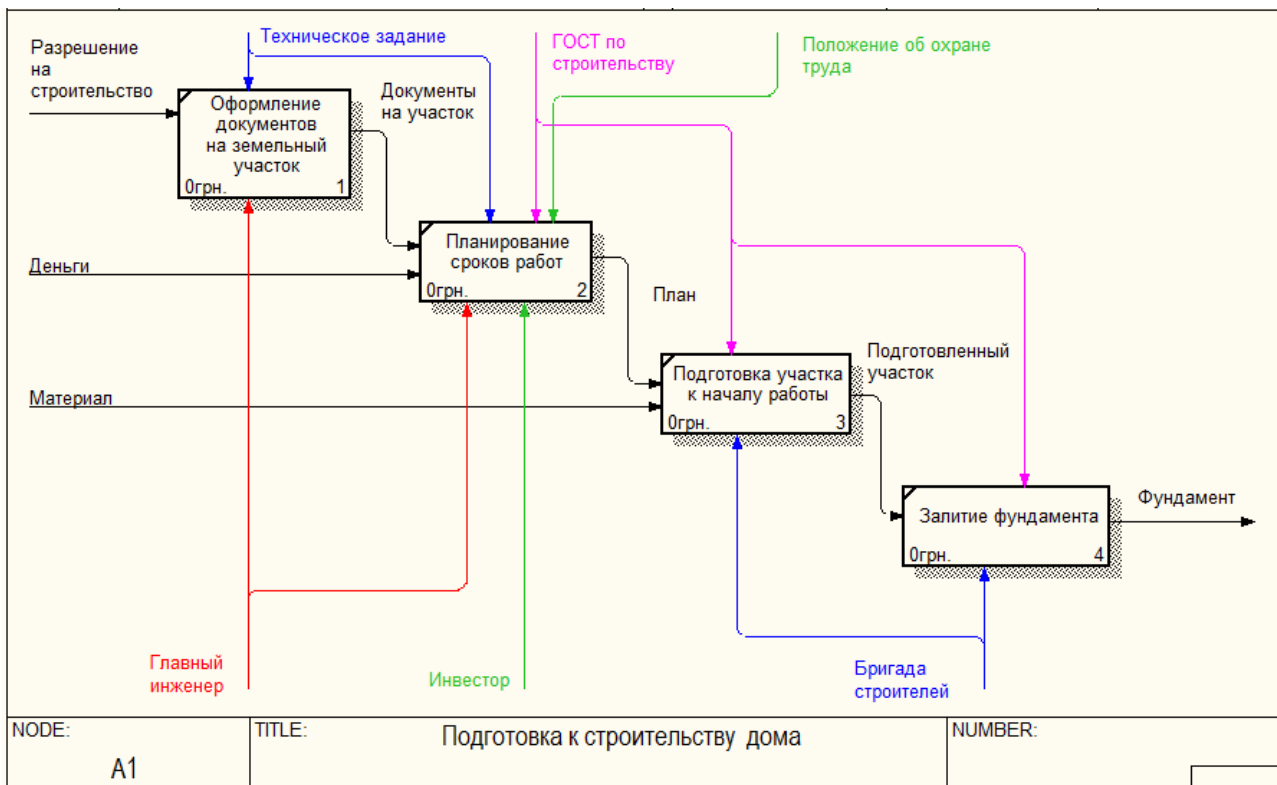


Рисунок 68 – Діаграма декомпозиції другого рівня процесу «Підготовка до побудови будинку»

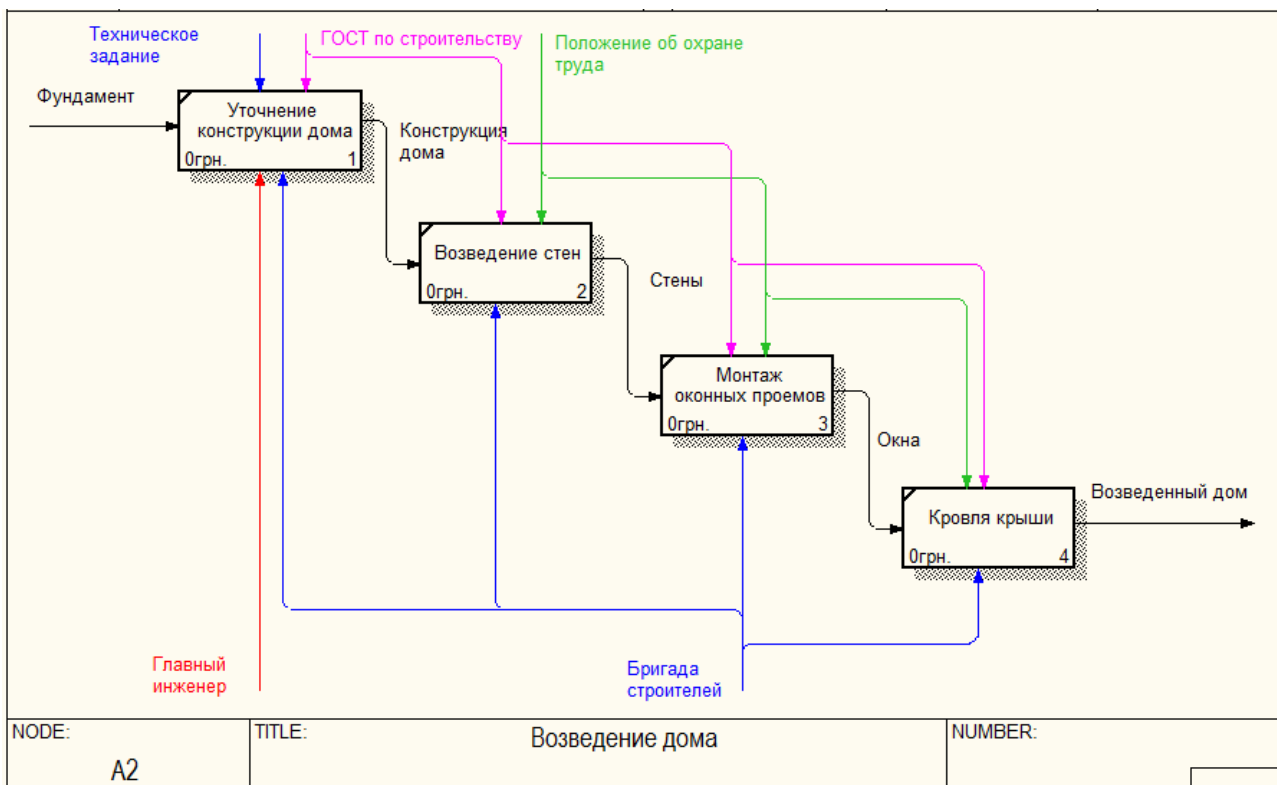


Рисунок 69 – Діаграма декомпозиції другого рівня процесу «Зведення будинку»

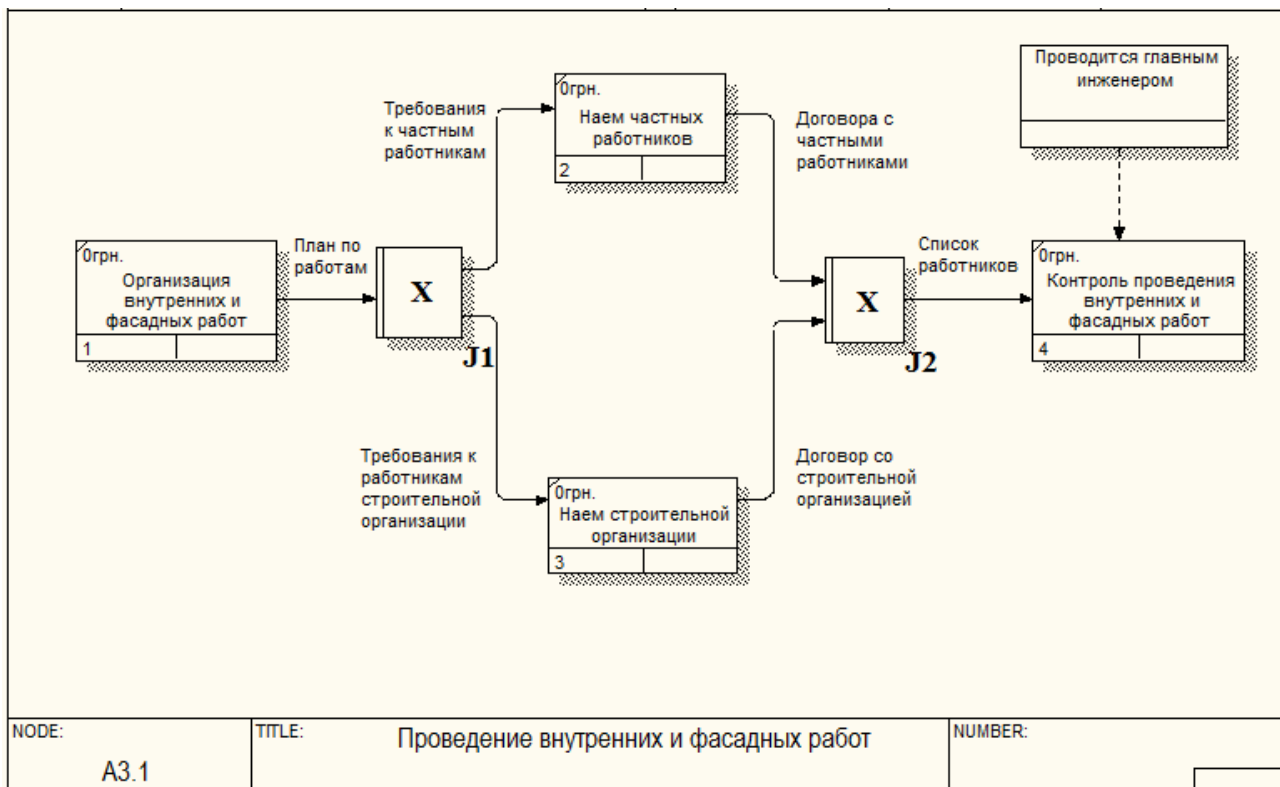


Рисунок 70 – Діаграма декомпозиції другого рівня процесу «Проведення внутрішніх та фасадних робіт»

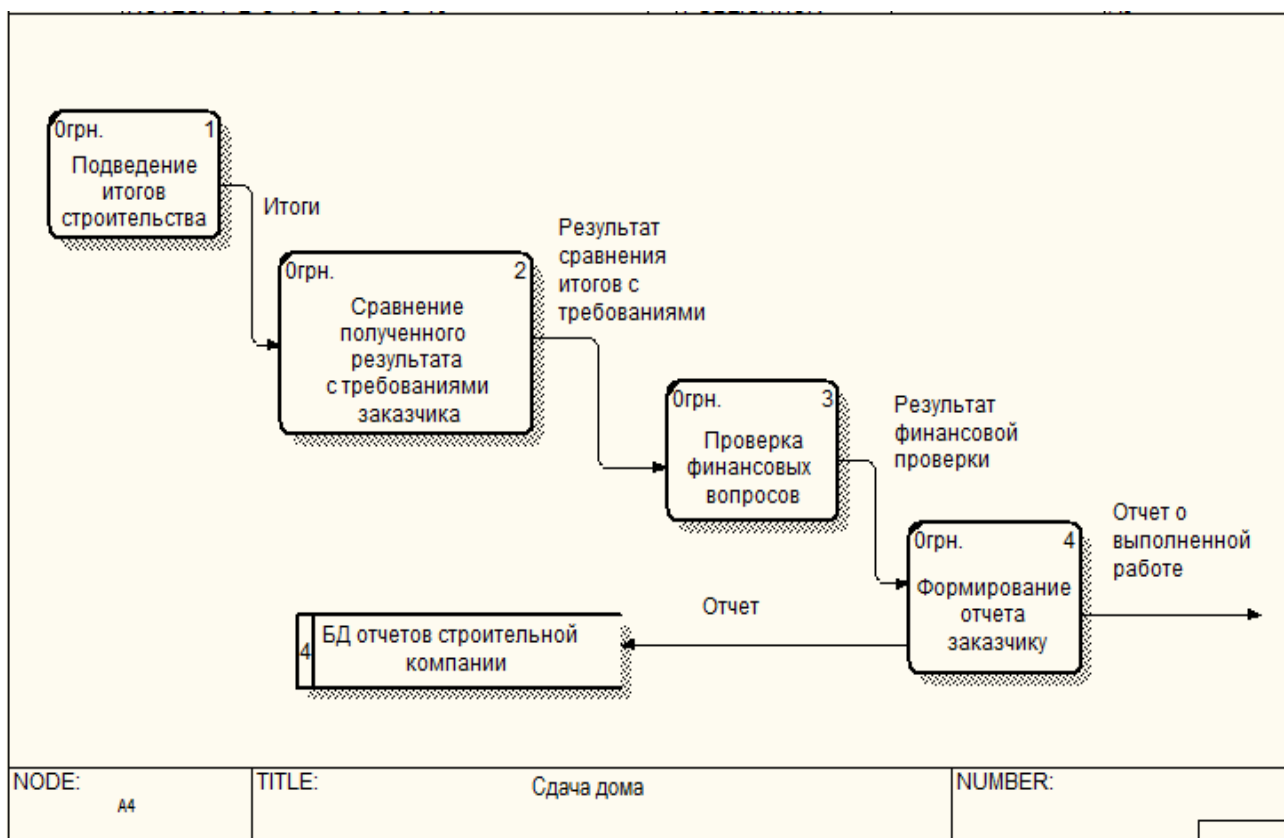


Рисунок 71 – Діаграма декомпозиції другого рівня процесу «Здача будинку»

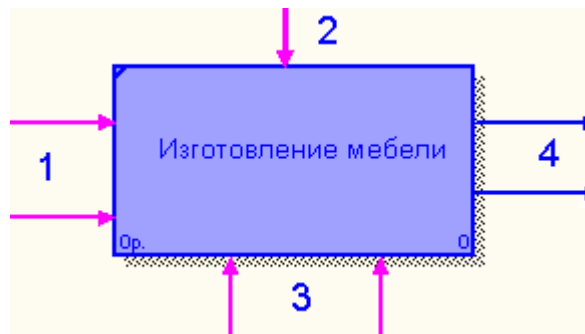
Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Контрольні питання

1. Що відображає контекстна діаграма в методологіях моделювання?
2. Що відображає діаграма декомпозиції в методологіях моделювання?
3. Назвіть основні поняття методології IDEF0.
4. Назвіть основні поняття методології DFD.
5. Назвіть основні поняття методології IDEF3.

Тестове завдання

1. Вкажіть, для чого призначена нижня сторона блоку (3):



- а) для управління (правила, стратегії, стандарти);
- б) для механізмів (ресурси, які виконують процес);
- в) для виходів (матеріал або інформація, що одержані в результаті виконання процесу);
- г) для входів (матеріал або інформація, що використовуються або перетворюються для отримання результату);
- д) правильні відповіді в), г).

2. Яка команда з контекстно-залежного меню дозволить змінити колір стрілки:

- а) Font Editor;
- б) Color Editor;
- в) Trim;
- г) Style Edition;
- д) Background Color;
- е) Squiggle.

3. IDFE0 – це:

- а) діаграма потоку даних;
- б) діаграма бізнес-процесу;
- в) діаграма сутність-зв'язок;
- г) діаграма ключів;
- д) всі відповіді правильні.

4. Скільки контекстних діаграм може бути в моделі:

- а) одна;
- б) дві;
- в) стільки, скільки рівнів деталізації;
- г) необмежену кількість;
- д) всі відповіді правильні.

5. DFD – це:

- а) діаграма об'єктів та процесів;
- б) діаграма потоку даних;
- в) контекстна діаграма;
- г) функціональна діаграма;
- д) діаграма сутність – зв'язок.

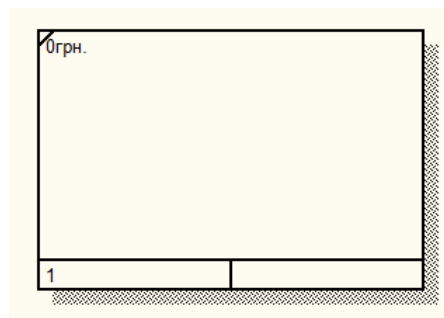
6. Які об'єкти описуються на діаграмі потоків даних:

- а) функції (роботи);
- б) сховища даних;
- в) зовнішні об'єкти;
- г) зовнішні посилання;
- д) всі відповіді правильні.

7. Що показують діаграми потоків даних?

- а) зовнішні джерела та приймачі даних;
- б) потоки даних;
- в) сховища (накопичувачі) даних;
- г) всі відповіді правильні;
- д) правильної відповіді немає.

8. В якій нотації використовується таке графічне зображення роботи:



- а) DFD;
- б) IDEF3;
- в) IDEF0;
- г) правильні відповіді б), в);
- д) правильні відповіді а), в).

9. За допомогою якого інструменту можна встановити між роботами перехрестя?

- а) ; б) ; в) ; г) ; д) .

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Вказівки з оформлення звіту:

1. Звіт оформлюється у текстовому редакторі Microsoft Word.
2. Звіт повинен містити:
 - назву лабораторної роботи;
 - мета лабораторної роботи;
 - формулювання завдань;
 - короткий опис ходу виконання завдань;
 - діаграми, що відображають результати виконання завдань;
 - висновки по роботі.

1.5 Ознайомлення з інструментальними засобами для опису процедур автоматизованої побудови мережевих моделей ГІС

Мета роботи: ознайомитися з інструментальними засобами для опису процедур автоматизованої побудови мережевих моделей ГІС.

Методичні вказівки по організації роботи студентів

Ознайомимося з інструментальним засобом для опису процедур автоматизованої побудови мережевих моделей ГІС – FPN (Fuzzy Petri Nets), нечіткі мережі Петрі (рис. 72).

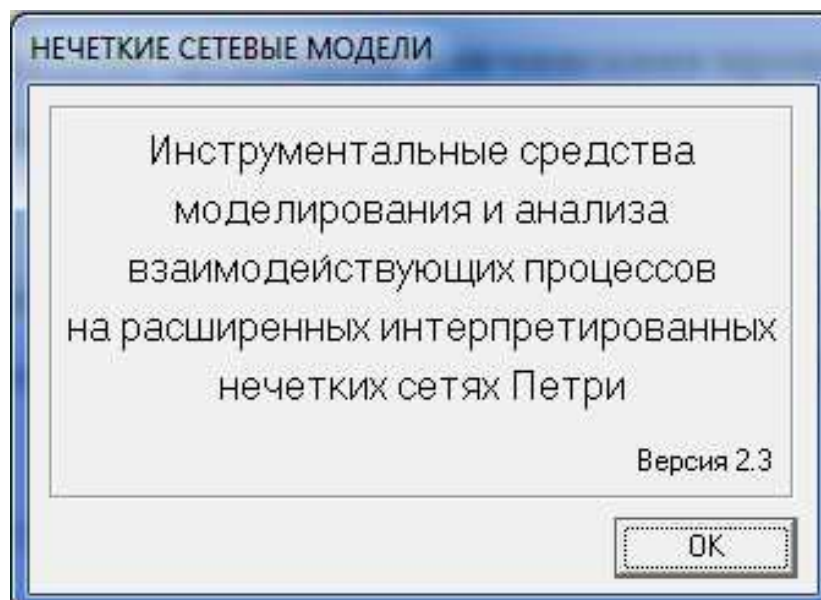


Рисунок 72 – Інструментальний засіб для опису процедур автоматизованої побудови мережевих моделей FPN

У даному інструментальному засобі для опису процедур автоматизованої побудови мережевих моделей ГІС надається ряд можливостей (рис. 73):

- побудова та аналіз;
- моделювання та аналіз;
- навчання;
- побудова;
- моделювання;
- аналіз.

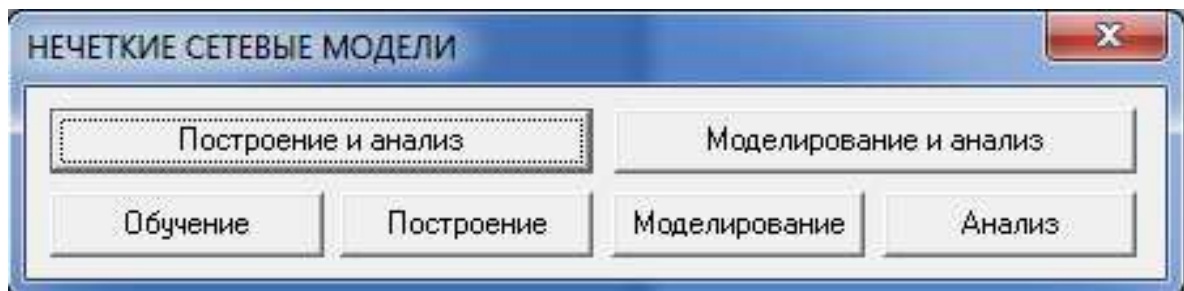


Рисунок 73 – Можливості інструментального засобу для опису процедур автоматизованої побудови мережевих моделей FPN

Важливою ланкою побудови сучасних систем виробничого призначення є створення ефективних моделей і засобів їх реалізації. Існує велика кількість моделей, які можуть бути використані в системах моделювання реальних процесів. Перспективним напрямом створення таких засобів є мережі Петрі (МП). Існуючі моделі мають меншу або рівну МП потужність моделювання. МП мають ряд суттєвих переваг, що робить їх перспективним апаратом моделювання складних процесів серед інших.

На ринку програмних продуктів існує ряд програм, що певною мірою дозволяють автоматизувати процеси, пов'язані з побудовою складних структур даних. Серед них слід зазначити Visual Case 6.0 фірми Rogue Wave Software, яка інтегрується в MS Visual Studio і є засобом візуального проектування структури програмного засобу. В основу функціонування цього засобу покладено можливість створення і перегляду існуючих зв'язків між окремими структурними елементами програми. Застосування цього продукту для моделювання програм викликає труднощі, програма забезпечує тільки візуалізацію класів, але не забезпечує їх моделювання та обробку.

Для моделювання складних процесів виникає потреба в автоматизованій процедурі побудови МП. Складність процедури візуального проектування МП полягає в динамічності структури даних, що відображає компоненти МП, а також у необхідності вирішення проблеми автоматичного розподілу пам'яті під кожен елемент уявлення МП.

Створення візуального інтерфейсу побудови МП надає можливість моделювати більш складні процеси. Процедура побудови МП була реалізована на мові MS Visual C++ 6.0 у вигляді програмного засобу PK_NET Graph Edition.

Основна частина програми, що реалізує саме уявлення МП складається з шести класів: CGENormalShema (блок-схема), CGELink (зв'язок), CGECtrlPoint (точка зв'язку), CGEPetriShema (вершина МП), CpetriShemaLink (зв'язок в МП), CHMatrix (матриця інцидентності).

Блок-схема складається з подання (CGENormalShema) і точок управління зв'язками (CGECtrlPoint). Кожна блок-схема має своє представлення у вигляді вершин і їх інцидентності в МП. Для забезпечення контрольованості уявлення блок-схеми в МП, клас CGENormalShema зберігає адреси відповідних їй вершин МП. Вершина МП статична і не дозволяє маніпуляцій зі зв'язками, тому не має в своєму складі контрольних точок.

Взаємодія окремих об'єктів між собою здійснюється за рахунок динамічного визначення предків класу та дозволяє редагувати модель. За рахунок цього організована інтерактивна операція зв'язку двох схем між собою через об'єкт зв'язку. Об'єкт-зв'язок є автономним об'єктом схеми, введений в структуру для можливості розташування точок зв'язку так, щоб він мав наочний вигляд.

Програма надає можливість автоматизувати процес побудови МП. Програма оснащена «дружнім інтерфейсом», який надає можливість користуватися програмним засобом користувачеві не програмісту.

Кожна блок-схема виведена на панель інструментів і є окремим елементом візуального відображення моделі. Для перегляду моделі у вигляді матриці інцидентності введено спеціальне подання у вигляді таблиці, що автоматично заповнюється при побудові моделі.

Програма забезпечує інтерактивну побудову моделі за допомогою звичайних блок-схем, їх вільне переміщення на віконному поданні.

Ефективність розробки підтверджена моделюванням реальних процесів.

Програмою запропоновано (рис. 74):

- панель опцій (1), що надає можливість керувати загальними можливостями програми;
- меню програми (2), що дублює кнопки швидкого доступу панелі опцій програми;
- панель конструктора моделі (3), за допомогою цієї панелі створюються моделі, що підлягають перетворенню в МП;
- віконне подання матриці інцидентності (4);
- подання області редагування моделі за допомогою блок-схем (5);
- подання області перегляду моделі у вигляді МП (6);
- рядок стану програми (7).

Створення моделі здійснюється за допомогою панелі конструктора. Щоб створити блок-схему, на панелі блок-схем слід натиснути ліву кнопку «миші», далі перемістити курсор «миші» у потрібне місце і знову натиснути ліву кнопку «миші».

Для переміщення елементів моделі слід натиснути ліву кнопку «миші» і, не відпускаючи її, перемістити курсор «миші» в потрібне місце. Щоб зафіксувати положення елемента моделі необхідно відпустити кнопку «миші».

Щоб створити логічний зв'язок між двома елементами моделі слід навести курсор «миші» на блакитну крапку, зображену на блок-схемі (представляє початок/кінець зв'язку), натиснути ліву кнопку «миші» і утримуючи її навести курсор «миші» на точку іншої блок-схеми та відпустити кнопку «миші». Після цього буде створено зв'язок двох схем, а на МП буде створено зв'язок двох вершин.

Для видалення блок-схеми з загального подання моделі слід натиснути праву кнопку «миші» над потрібної блок-схемою і з меню вибрати пункт «Видалити», після цього блок-схема і її подання на МП буде видалено з моделі.

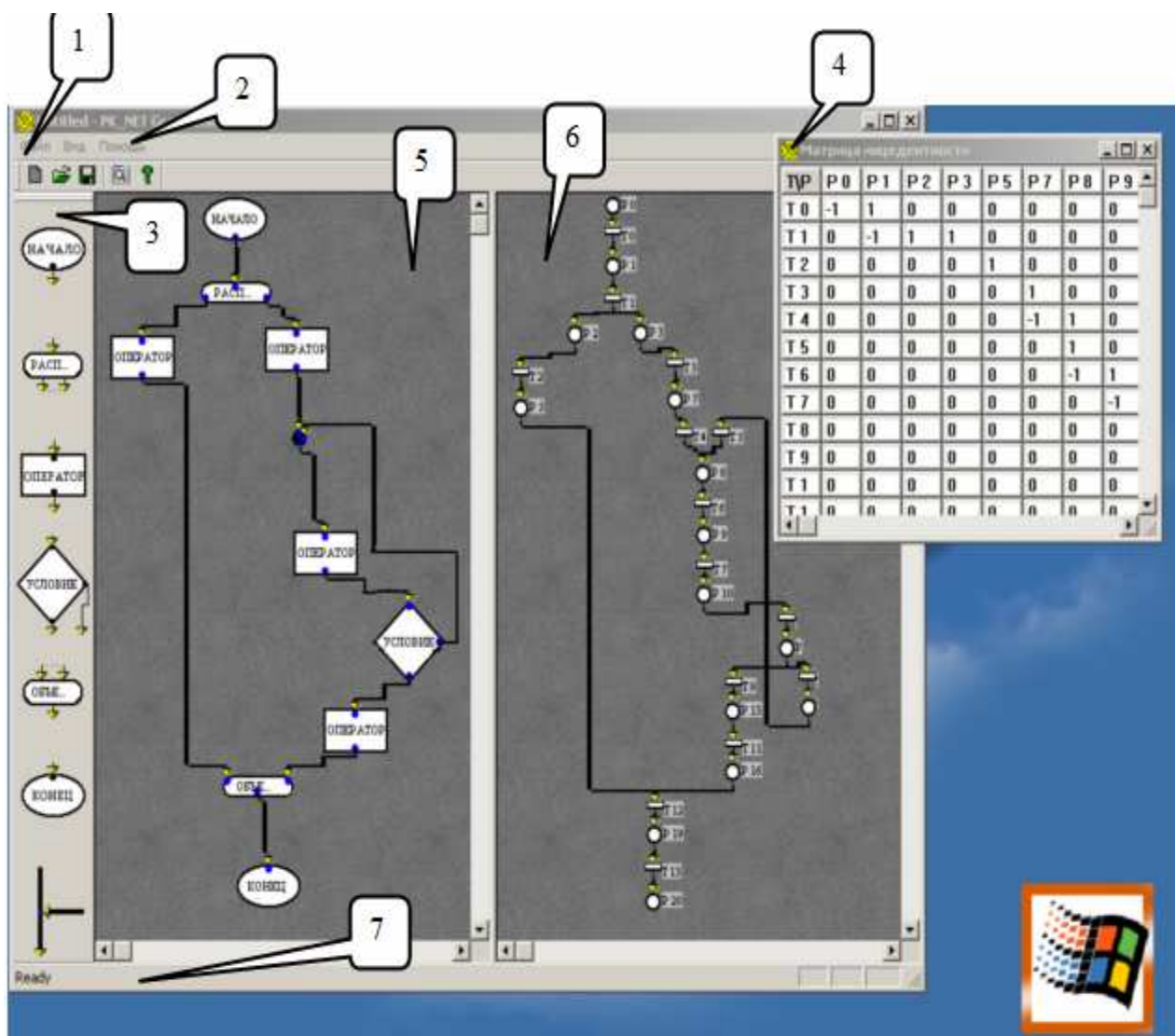


Рисунок 74 – Можливості інструментального засобу для опису процедур автоматизованої побудови мережевих моделей FPN

Для вилучення елемента зв'язку між двома блок-схемами слід навести курсор «миші» на потрібний зв'язок і, натиснувши праву кнопку «миші», вибрати пункт меню «Видалити».

Зміна зв'язків забезпечується за рахунок вилучення зв'язку з моделі і відповідно створення нового зв'язку з іншої блок-схемою. У процесі зміни зв'язків змінюється матриця інцидентності та подання МП.

Щоб отримати інтерактивну підказку слід навести курсор «миші» на потрібний об'єкт програми, через 1-2 секунди на екрані буде відображена підказки для цього об'єкта.

Користувач має можливість зберегти інформацію щодо моделі у вигляді файлів *.Mtp – матриці інцидентності і *.Nmr – вектора початкової маркірування за допомогою меню «Файл – Зберегти як».

Інтерпретація мережевих моделей подана на рисунку 75.

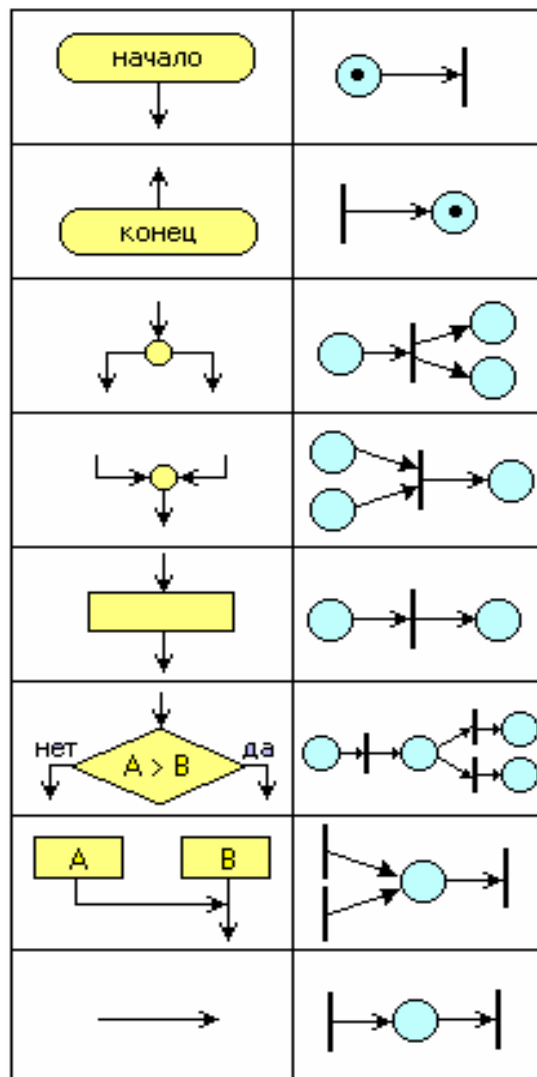


Рисунок 75 – Інтерпретація мережевих моделей

Приклад моделі подано на рисунку 76.

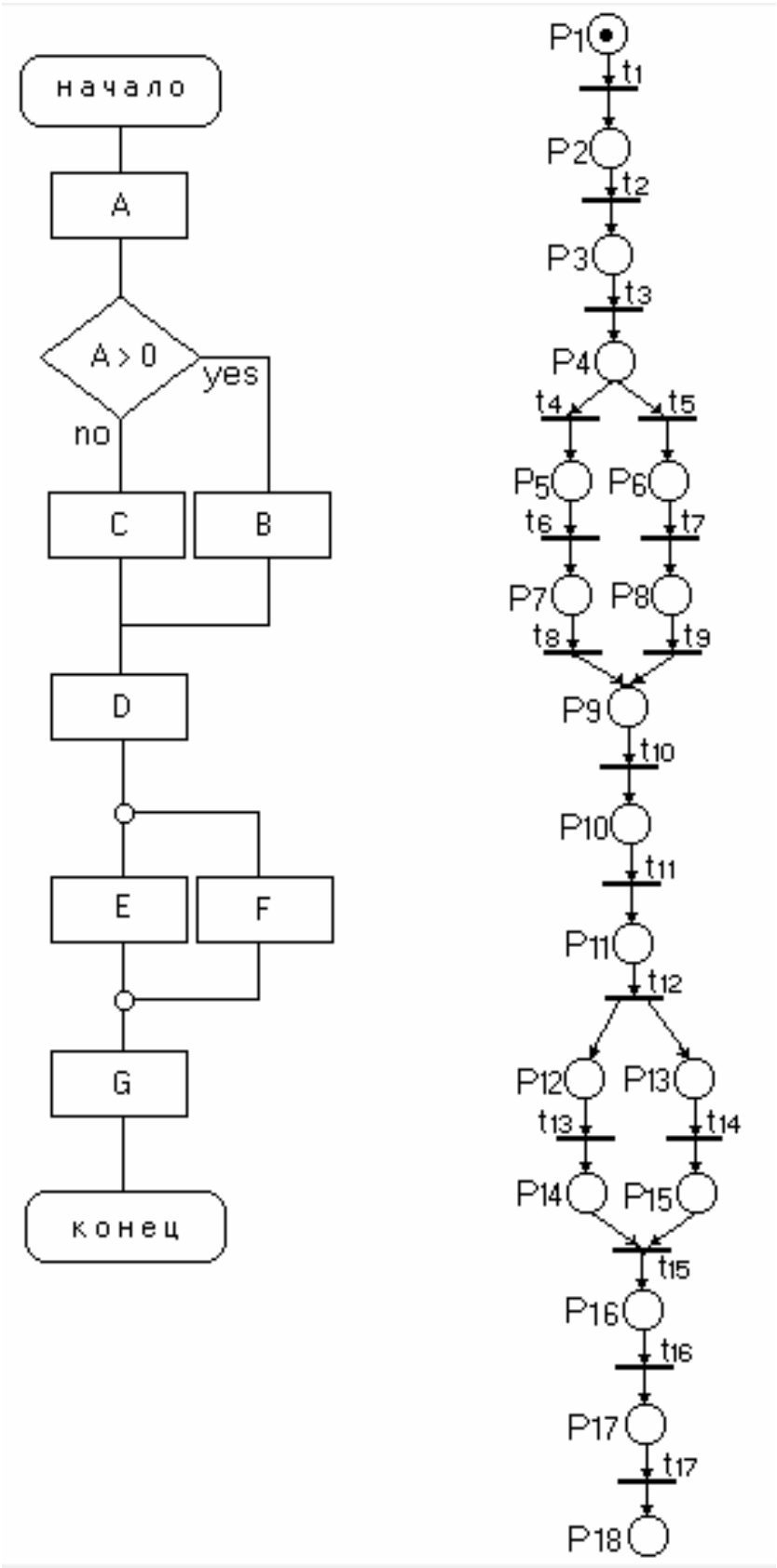


Рисунок 76 – Приклад моделі

Алгоритм моделі (рис. 76) буде мати такий вигляд:

Крок 1. Початок.

Крок 2. Реалізація процесу А.

Крок 3. Виконання умови: якщо виконується умова $A > 0$, то переходимо до кроку 4, інакше – до кроку 5.

Крок 4. Реалізація процесу В.

Крок 5. Реалізація процесу С.

Крок 6. Виконання розгалуження: реалізація процесу Д через розбиття на підпроцес Е та підпроцес Ф.

Крок 7. Реалізація підпроцесу Е.

Крок 8. Реалізація підпроцесу Ф.

Крок 9. Виконання злиття: реалізація процесу Г на основі результатів підпроцесу Е та підпроцесу Ф.

Крок 10. Кінець.

Завдання. Реалізувати в інструментальному програмному засобі моделювання взаємодіючих процесів на розширених інтерпретованих нечітких мережах Петрі свій приклад процесу.

Обмеження. Побудувати фрагмент моделі з обраної тематики, використовуючи всі типи елементів блок-схеми (рис. 75). Інтерпретувати фрагмент моделі в мережеву модель за допомогою інструментального засобу моделювання взаємодіючих процесів на розширених інтерпретованих нечітких мережах Петрі (рис. 75). Описати алгоритм.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Контрольні питання

1. Які види мереж Петрі Ви знаєте?
2. З яких елементів складається мережа Петрі?
3. Назвіть основні властивості мереж Петрі.
4. Охарактеризуйте правила побудови мережевих моделей.
5. Як виконується дослідження збереженості мережі Петрі?
6. Як виконується дослідження досяжності маркірування мережі Петрі?
7. Як визначається рівень активності переходу мережі Петрі?
8. Що називають деревом досяжності мережі Петрі?
9. Як складається дерево досяжності мережі Петрі?
10. Опишіть алгоритм складання дерева досяжності.

Тестове завдання

1. Мережею називається:

- а) зв'язний граф, в якому задані «пропускні здатності» ребер;
- б) зв'язний граф, в якому задані «пропускні здатності» вершин.

2. Розвиток мереж Петрі розпочався у:

- а) 1952 році;
- б) 1962 році;
- в) 1972 році.

3. Мережа Петрі називається «безпечною» якщо:

- а) кожна позиція містить не більше однієї мітки;
- б) кількість міток в кожній позиції не перевищує деяке ціле число k ;
- в) загальна кількість міток у всіх позиціях завжди постійна.

4. Мережа Петрі складається з:

- а) двох елементів;
- б) трьох елементів;
- в) чотирьох елементів.

5. Мережі Петрі є математичною моделлю в термінах:

- а) «умова – подія»;
- б) «умова – умова»;
- в) «подія – подія».

6. Перехід мережі Петрі позначається:

- а) кругом;
- б) вертикальною планкою;
- в) дугою.

Після того, як ця робота буде виконана, запросіть викладача і продемонструйте йому результат.

Вказівки з оформлення звіту:

1. Звіт оформлюється у текстовому редакторі Microsoft Word.

2. Звіт повинен містити:

- назву лабораторної роботи;
- мета лабораторної роботи;
- формулювання завдань;
- короткий опис ходу виконання завдань;
- діаграми, що відображають результати виконання завдань;
- висновки по роботі.

2 ВКАЗІВКИ ДО ВИКОНАННЯ РОЗРАХУНКОВО-ГРАФІЧНИХ РОБІТ

Розрахунково-графічні роботи № 1 та № 2 є складовими частинами навчальної дисципліни «Основи моделювання складних систем» та призначені для практичного закріплення і розширення отриманих теоретичних знань.

Розрахунково-графічне завдання № 1 виконується в межах теми «Особливості застосування інструментальних засобів для побудови моделей геоінформаційних систем» – 18 годин. Розрахунково-графічне завдання № 2 виконується в межах теми «Застосування інструментальних програмних засобів для моделювання об'єктів та процесів геоінформаційних систем» – 18 годин.

Кожний студент виконує індивідуальне завдання, яке видається йому викладачем.

2.1 Структура розрахунково-графічних робіт

За результатами розрахунково-графічної роботи оформлюється пояснювальна записка і здається викладачеві на перевірку. Пояснювальна записка оформлюється згідно до ДСТУ 3008-95 та ДСТУ ГОСТ 7.1:2006 і повинна містити:

- вступ;
- постановку завдання дослідження;
- аналіз предметної області;
- синтез графічної моделі досліджуваного процесу;
- формування стратегії поліпшення проблемного процесу;
- висновок;
- список використаної літератури.

У вступі необхідно дати короткий опис предметної області, сформулювати мету роботи і показати актуальність проведеного дослідження.

Постановка завдання дослідження повинна містити:

- визначення сфери діяльності;
- виявлення нормативних документів, що встановлюють вимоги до процесу;
- встановлення учасників процесу і структури їх підпорядкованості.

У розділі «Аналіз предметної області» необхідно виявити найбільшу кількість особливостей процесу та їх зв'язок з проблемною областю, описати алгоритм і принципи його функціонування із зазначенням переліку окремих підсистем, їх функцій і цілей.

Розділ «Аналіз предметної області» може містити таку інформацію:

- встановлення складу вхідної, вихідної та внутрішньої документації процесу;
- опис порядку функціонування процесу;
- аналіз зв'язку процесу з досліджуваною проблемою.

У розділі «Синтез графічної моделі досліджуваного процесу» необхідно сформувати загальне графічне представлення досліджуваного процесу.

Подається вербальний і графічний опис функціонування процесу, що включає графічну схему виконання процесу і текстові коментарі, що пояснюють на схемі як виконується процес.

У розділі «Формування стратегії поліпшення проблемного процесу» необхідно сформувати пропозиції щодо вдосконалення процесу.

У висновку повинні бути підведені підсумки виконаної роботи, вказується обсяг виконаної роботи, яка вирішена задача і практична користь, яку принесло її рішення.

Список використаної літератури повинен містити посилання на основні джерела, використані при виконанні розрахунково-графічної роботи.

2.2 Правила оформлення розрахунково-графічних робіт

Розрахунково-графічну роботу оформлюють на аркушах формату А4 (210x297 мм). Текст роботи слід друкувати, дотримуючись таких розмірів полів: верхнє, ліве і нижнє – не менше 20 мм, праве – не менше 10 мм.

Гарнітура шрифту роботи – Times New Roman (Сур), кегль – 14, колір друку – чорний, міжрядковий інтервал – полуторний. Щільність тексту повинна бути рівномірною (без розріджень і ущільнень). Абзацний відступ повинен бути однаковим впродовж усього тексту і дорівнювати п'яти знакам.

Розділи і підрозділи повинні мати заголовки. Заголовки структурних елементів роботи і заголовки розділів слід розташовувати посередині рядка і друкувати великими літерами без крапки в кінці, не підкреслюючи.

Нижче кожного розділу повинно бути залишено не менше одного вільного рядка. Вище і нижче кожного підрозділу повинно бути залишено не менше одного вільного рядка.

Заголовки підрозділів, пунктів і підпунктів слід починати з абзацного відступу і друкувати маленькими літерами, крім першої великої, не підкреслюючи, без крапки в кінці. Якщо заголовок складається з двох або більше речень, їх розділяють крапкою.

Сторінки роботи слід нумерувати арабськими цифрами, дотримуючись наскрізної нумерації по всьому тексту. Номер сторінки проставляється в правому верхньому кутку сторінки без крапки в кінці. Титульний аркуш включають до загальної нумерації сторінок звіту. Номер сторінки на титульному листі не проставляють.

Розділи, підрозділи, пункти, підпункти роботи слід нумерувати арабськими цифрами. Розділи роботи повинні мати порядкову нумерацію в межах викладення суті роботи і позначатися арабськими цифрами без крапки, наприклад, 1, 2, 3...

Підрозділи повинні мати порядкову нумерацію в межах кожного розділу. Номер підрозділу складається з номера розділу і порядкового номера підрозділу, розділених крапкою. Після номера підрозділу крапку не ставлять, наприклад, 1.1, 1.2 і т. д.

Пункти повинні мати порядкову нумерацію в межах кожного розділу або підрозділу. Номер пункту складається з номера розділу і порядкового номера пункту, або з номера розділу, порядкового номера підрозділу та порядкового номера пункту, відокремлених крапкою. Після номера пункту крапку не ставлять, наприклад, 1.1, 1.2 або 1.1.1, 1.1.2 і т. д.

Якщо розділ або підрозділ складається з одного пункту, або пункт складається з одного підпункту, його нумерують.

Ілюстрації (креслення, рисунки, графіки, схеми, діаграми, фотознімки) слід розташовувати в роботі безпосередньо після тексту, в якому вони згадуються вперше, або на наступній сторінці. На всі ілюстрації мають бути посилання у роботі. Якщо ілюстрації створені не автором звіту, то необхідно дотримуватися вимог чинного законодавства про авторські права.

Ілюстрації можуть мати назву, яку розміщують під ілюстрацією. При необхідності під ілюстрацією розміщують пояснювальні дані (під рисунковий текст). Ілюстрація позначається словом «Рисунок __», яке разом з назвою ілюстрації поміщають після пояснювальних даних, наприклад, «Рисунок 3.1 – Схема розміщення». Ілюстрації слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу, за винятком ілюстрацій, наведених у додатках. Номер ілюстрації складається з номера розділу і порядкового номера ілюстрації, розділених крапкою, наприклад, рисунок 3.2 – другий рисунок третього розділу.

Цифровий матеріал, як правило, оформлюють у вигляді таблиць. Таблицю слід розташовувати безпосередньо після тексту, в якому вона згадується вперше, або на наступній сторінці.

На всі таблиці повинні бути посилання в тексті звіту.

Таблиці слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу, за винятком таблиць, що наводяться в додатках. Номер таблиці складається з номера розділу і порядкового номера таблиці, відокремлених крапкою, наприклад, таблиця 2.2 – перша таблиця другого розділу. Таблиця може мати назву, яку друкують малими літерами (крім першої великої) і вміщують над таблицею. Назва повинна бути короткою і відбивати зміст таблиці.

Напис «Таблиця 2.2 – Назва таблиці» вказують один раз зліва над першою частиною таблиці, над іншими пишуть: «Продовження таблиці 2.2» із зазначенням номера таблиці.

Заголовки граф таблиці друкують з великих літер, а підзаголовки – з малої, якщо вони складають одне речення із заголовком.

Перерахування, при необхідності, можуть бути приведені всередині пунктів або підпунктів. Перед перерахуванням ставлять двокрапку.

Перед кожною позицією переліку слід ставити малу літеру українського алфавіту з дужкою, або, не нумеруючи – дефіс (перший рівень деталізації).

Для подальшої деталізації переліку слід використовувати арабські цифри з дужкою (другий рівень деталізації).

Перерахування першого рівня деталізації друкують малими літерами з абзацного відступу, другого рівня – з відступом щодо місця розташування перерахувань першого рівня.

Формули та рівняння розташовують безпосередньо після тексту, в якому вони згадуються, посередині сторінки. Вище і нижче кожної формули або рівняння повинно бути залишено не менше одного вільного рядка.

Формули та рівняння слід нумерувати порядковою нумерацією в межах розділу. Номер формули або рівняння зазначають на рівні формули або рівняння в дужках у крайньому правому положенні на рядку.

Посилання в тексті на джерела слід зазначати порядковим номером за переліком посилань, виділеним двома квадратними дужками. Бібліографічні описи в переліку посилань подають у порядку, в якому вони вперше згадуються в тексті. Бібліографічні описи в переліку посилань подають на мові оригіналу.

2.3 Приклади бібліографічного опису в переліку посилань

Книга одного автора:

Андреев, В. В. Как организовать делопроизводство на предприятии [Текст] / В. В. Андреев. – М.: ИНФРА-М, 1997. – 94 с.

Книга двох авторів:

Белов, А. В. Финансы и кредит [Текст]: учеб. / А. В. Белов, В. Н. Николаев. – К.: Университет, 2004. – 215 с.

Книга трьох авторів:

Агафонова, Н. Н. Гражданское право [Текст]: учеб. пособие / Н. Н. Агафонова, Т. В. Богачева, Л. И. Глушкова; под общ. ред. А. Г. Калпина. – Х.: Фактор, 2000. – 542 с.

Книга чотирьох авторів:

Элементы информатики [Текст]: довідник / В. С. Височанський, А. І. Кардаш, В. С. Костев, В. В. Черняхівський. – К.: Наук. думка, 2003. – 192 с.

Книга п'яти авторів та більше:

Коротковолновые антенны [Текст]: учеб. пособие / Г. З. Айзенберг, С. П. Белоусов, Я. М. Журбин и др.; под общ. ред. А. А. Стогния. – М.: Радио и связь, 2003. – 192 с.

Перекладне видання:

Нойман, Э. Происхождение и развитие сознания [Текст]: пер. с англ. – К.: Ваклер, 1998. – 462 с.

Книги під заголовком:

Информационные технологии в маркетинге [Текст]: учеб. / под ред. Г. А. Титаренко. – М.: ЮНИТИ, 2000. – 335 с.

Статті із журналів:

Гончаров, В. А. Численная схема моделирования дозвуковых течений вязкого сжимаемого газа [Текст] / В. А. Гончаров, В. М. Кравцов // Журн. вычисл. математики и мат. физики. – 1988. – Т. 28, №12. – С. 1858-1866.

Збірники наукових праць:

Отчет о выполнении плана научно-исследовательских работ за 2003 год [Текст]: сб. науч. тр. / Рос. Акад. мед. наук, Сиб. отд. – Новосибирск: СО РАМН, 2004. – 83 с.

Тези конференцій:

Образование, наука, производство на пути углубления интеграции и повышения качества инженерного образования [Текст]: тез. докл. науч.-практ. конф. (окт. 2000) / отв. ред. В. Г. Вдовенко. – Красноярск: САА, 2000. – 53 с.

Матеріали конференцій:

Проблемы экономики, организации и управления реструктуризацией и развитием предприятий промышленности [Текст]: материалы IV междунар. науч.-практ. конф., 30 марта 2005 г. Новочеркасск / редкол.: Б. Ю. Серебряков (отв. ред.). – Новочеркасск: Темп, 2005. – 58 с.

Дисертації:

Антопольский, А. Б. Описание информационных языков [Текст]: дис. ... канд. филол. наук / А. Б. Антопольский. – М., 1969. – 404 с.

Автореферати дисертацій:

Бутковский, О. Я. Обратные задачи хаотичной динамики и проблемы предсказуемости хаотичных процессов [Текст]: автореф. дис. ... д-ра физ.-мат. наук: 01.04.03 / О. Я. Бутковский; [Ин-т радиотехники и электроники РАН]. – М., 2004. – 39 с.

Віддаленого доступу:

Основные направления исследований, основанные на семантическом анализе текстов [Электронный ресурс] / С.-Петербург. гос. ун-т, фак. прикладной математики – процессов управления. – Режим доступа: [www/ URL: http://arcp.arpmath.spbu.ru/ru/staff/tuzov/onapr.html/](http://arcp.arpmath.spbu.ru/ru/staff/tuzov/onapr.html/) – 10.12.2004 г. – Загл. с экрана.

Локального доступу:

Internet шаг за шагом [Электронный ресурс]: интеракт. учеб. – Электрон. дан. и прогр. – СПб.: Питер Ком, 1997. – 1 электрон. опт. диск (CD-ROM). – Систем. требования: ПК от 486 DX 66 МГц; RAM 1616 Мб; Windows 95; зв. плата. – Загл. с этикетки диска.

2.4 Критерії оцінювання розрахунково-графічних робіт

Критеріями оцінювання розрахунково-графічних робіт є:

- глибина і актуальність знань, показаних студентом в розрахунково-графічній роботі;
- відповідність отриманих результатів завданню.

Оцінка «зараховано» (з подальшим перерахунком у відповідні бали) ставиться, якщо всі критерії повністю досягнуті студентом.

Оцінка «не зараховано» ставиться, якщо не повністю досягнуті всі критерії або вчинено більше 2 грубих помилок, у такому разі робота повертається студентові на доопрацювання. Якщо хоча б одну розрахунково-графічну роботу не зараховано, то студент не може бути допущений до екзамену з дисципліни «Основи моделювання складних систем».

3 ВКАЗІВКИ ДО ВИКОНАННЯ САМОСТІЙНИХ РОБІТ

Завданням самостійної роботи студентів є опрацювання додаткової інформації для більш поглибленого вивчення дисципліни «Основи моделювання складних систем»:

- побудова та дослідження математичних моделей взаємодіючих динамічних процесів;
- побудова та дослідження імітаційних моделей взаємодіючих динамічних процесів;
- побудова та дослідження мережевих моделей ГІС;
- дослідження мережевих моделей ГІС за критеріями досяжності;
- моделювання ГІС під час вибору альтернатив в рішеннях, що приймаються.
- дослідження моделей подання знань в знання орієнтованих системах
- побудова моделей фрагментів баз знань та дослідження особливостей алгоритмів логічного виведення на знаннях

3.1 Основні поняття нечіткості

Теорія нечітких множин бере свій початок з 1965 року, коли професор Лотфі Заде з університету Берклі в США опублікував свою роботу «Fuzzy Sets» у журналі «Information and Control».

Прикметник «fuzzy» (нечіткий, розмитий) введено в назву нової теорії з метою відокремлення від традиційної чіткої математики, що оперує з чіткими поняттями: «належить – не належить», «істина – хибність».

Початком практичного застосування теорії нечітких множин можна вважати 1975 рік, коли Е. Мамдані побудував перший нечіткий контролер, заснований на нечітких лінгвістичних правилах «якщо – то», що привело до підвищення інтересу до теорії нечітких множин.

У 1992 році Коско була доведена *теорема про нечітку апроксимацію*, відповідно до якої будь-яка математична система може бути апроксимована системою, заснованою на нечіткій логіці. Іншими словами, за допомогою природно-мовних висловлень-правил «якщо – то», з подальшою їх формалізацією засобами теорії нечітких множин, можна точно відбити довільний взаємозв'язок «вхід – вихід».

У 1992 році Ванг показав, що нечітка система є універсальним апроксиматором, тобто може апроксимувати будь-яку неперервну функцію з довільною точністю, якщо використовувати набір з n ($n \rightarrow \infty$) правил виду «якщо – то», гаусові функції належності, композиції у вигляді добутку, імплікації у формі Ларсена та центроїдний метод приведення до чіткості.

У 1995 році Кастро показав, що логічний контролер Мамдані також є універсальним апроксиматором при симетричних трикутних функціях належності, композиції з використанням операції мінімуму, імплікації у формі Мамдані і центроїдного методу приведення до чіткості.

Системи з нечіткою логікою **доцільно застосовувати** для складних процесів, коли експертні знання про об'єкт або про процес можна сформулювати тільки в лінгвістичній формі.

Системи, що базуються на нечіткій логіці, **застосовувати недоцільно**, якщо необхідний результат може бути отриманий яким-небудь іншим (стандартним) шляхом, або якщо для об'єкта або процесу вже знайдена адекватна і легко досліджувана математична модель.

Основні недоліки систем з нечіткою логікою:

- вихідний набір нечітких правил, що постулюються, формулюється експертом-людиною і може виявитися неповним або суперечливим;
- вид і параметри функцій належності, що описують вхідні і вихідні змінні системи, вибираються суб'єктивно і можуть виявитися такими, що цілком не відбивають реальну дійсність.

Нехай U – універсальна множина, u – елемент U , а G – деяка властивість. Звичайна (чітка) підмножина A універсальної множини U , елементи якої мають властивість G , визначається як множина впорядкованих пар $\{ \langle \mu_A(u) | u \rangle \}$, де $\mu_A(u)$ – характеристична функція належності, що приймає значення 1, якщо u має властивість G , та 0 – у протилежному випадку.

Нечітка підмножина відрізняється від звичайної тим, що для елементів u з U немає однозначної відповіді «ні» або «так» щодо властивості G .

У зв'язку з цим **нечітка підмножина** A універсальної множини U визначається як множина впорядкованих пар $A = \{ \langle \mu_A(u) | u \rangle \}$, де $\mu_A(u)$ – характеристична *функція належності*, що приймає значення в деякій цілком впорядкованій множині M (наприклад, $M = [0; 1]$).

Функція належності вказує ступінь належності елемента u нечіткій підмножині A . Множину M називають *множиною належностей*.

Якщо $M=\{0, 1\}$, то нечітка підмножина A може розглядатися як чітка множина. Чітку множину A можна розглядати як граничний випадок нечіткої множини A , функція належності якої $\mu_A(u)$ набуває лише бінарних значень.

Приклад: подати у вигляді нечіткої множини поняття «Чоловік середнього зросту» на універсальній множині $U=\{155, 160, 165, 170, 175, 180, 185, 190\}$. Одне з можливих рішень виглядає так: $A=(0/155, 0,1/160, 0,3/165, 0,8/170, 1/175, 1/180, 0,5/185, 0/190)$.

Нечітка змінна визначається як $\langle a, U, A \rangle$, де a – найменування змінної, $U=\{u\}$ – область визначення змінної, $A=\{\langle \mu_A(u)|u \rangle\}$ – нечітка множина, що описує обмеження на можливі значення змінної a .

Нечітка змінна – це теж саме, що і нечітке число, тільки з додаванням імені, яким формалізується поняття, що описується цим числом. Для людини зручніше задавати значення змінної не числами, а словами. Щодня ми приймаємо рішення на основі лінгвістичної інформації такого типу: «дуже висока температура», «утомлива поїздка», «швидка відповідь», «красивий букет», «гармонійний смак» і тому подібне. Психологи встановили, що в людському мозку майже вся числова інформація вербально перекодовується і зберігається у вигляді слів.

Лінгвістична змінна – це множина нечітких змінних, вона використовується для того, щоб дати словесний опис деякому нечіткому числу, отриманому в результаті деяких операцій.

Лінгвістична змінна визначається як (3.1)

$$\langle x, L, U, G, M \rangle, \quad (3.1)$$

де x – найменування змінної;

L – множина її значень, що складається з найменувань нечітких змінних, областю визначення кожної з яких є множина U ;

G – синтаксична процедура, що дозволяє оперувати елементами терм-множини L , зокрема, генерувати нові осмислені терми;

M – семантична процедура, що дозволяє приписати кожному новому значенню лінгвістичної змінної нечітку семантику, шляхом формування нової нечіткої множини.

Терм-множина – це множина всіх можливих значень лінгвістичної змінної.

Терм – будь-який елемент терм-множини. У теорії нечітких множин терм формалізується нечіткою множиною за допомогою функції належності.

Наприклад, змінна «швидкість автомобіля» може набувати значень «низька», «середня», «висока» і «дуже висока». В цьому випадку лінгвістичною змінною є «швидкість автомобіля», термами – лінгвістичні оцінки «низька», «середня», «висока» і «дуже висока», які і складають терм-множину.

Нечіткий терм – це нечітка множина, яка має властивість, якій відповідає певне поняття.

Недетермінованість висновків означає, що заздалегідь шлях вирішення конкретної задачі в просторі її станів визначити неможливо. Тому в більшості випадків методом проб і помилок вибирається деякий ланцюжок логічних висновків, що узгоджуються з наявними знаннями, а у випадку, якщо він не приводить до успіху, то організовується перебір з поверненням для пошуку іншого ланцюжка. Такий підхід припускає визначення деякого первісного шляху. Недетермінованість висновків варто враховувати при розробці ефективних способів подання і збереження знань, а також при побудові методів пошуку й обробки знань, що дозволяють одержати вирішення задачі за найменше число кроків.

Багатозначність інтерпретації – звичайне явище в задачах розпізнавання. При розумінні природної мови серйозними проблемами стають багатозначність змісту слів, їхня підпорядкованість, порядок слів у реченні. Проблеми розуміння змісту виникають у будь-якій системі, що взаємодіє з користувачем природною мовою. Розпізнавання графічних образів також пов'язано з вирішенням проблеми багатозначної інтерпретації. При комп'ютерній обробці знань багатозначність необхідно усувати шляхом вибору правильної інтерпретації.

Ненадійність знань і висновків означає, що для оцінки вірогідності знань не можна застосувати двобальну шкалу (1 – абсолютно достовірні знання, 0 – недостовірні знання). Для більш тонкої оцінки вірогідності знань застосовується імовірнісний підхід, заснований на теоремі Байєса. Широке застосування на практиці одержали нечіткі висновки, які будуються на базі нечіткої логіки.

Неповнота знань і немонотонна логіка. Абсолютно повних знань не буває, оскільки процес пізнання нескінченний. У зв'язку з цим стан бази знань повинний змінюватися з часом. На відміну від простого додавання інформації, як у базах даних, при додаванні нових знань виникає небезпека одержання суперечливих висновків: тобто висновки, отримані з використанням нових знань, можуть спростовувати ті, що були отримані раніше. Ще гірше, якщо нові

знання будуть знаходитися в протиріччі зі старими, тоді механізм висновку може стати непрацездатним.

Більшість експертних систем першого покоління були засновані на моделі закритого світу, обумовленій застосуванням апарату формальної логіки для обробки знань.

Модель закритого світу припускає твердий добір знань, що включаються в базу, а саме: база знань заповнюється винятково вірними поняттями, а усе, що ненадійно або невиразно, свідомо вважається помилковим. Така модель має обмежені можливості подання знань.

Недоліки моделі закритого світу пов'язані з тим, що формальна логіка виходить з передумови, відповідно до якої набір визначених у системі аксіом є *повним* (теорія є повною, якщо кожний її факт можна довести, виходячи з аксіом цієї теорії). Для повного набору знань справедливність раніше отриманих висновків не порушується з додаванням нових фактів. Ця властивість логічних висновків називається *монотонністю*.

Неточність знань. Відомо, що кількісні дані можуть бути неточними, при цьому існують кількісні оцінки такої неточності. Лінгвістичні знання також можуть бути неточними. Для врахування неточності лінгвістичних знань використовується теорія нечітких множин. Фактично нечіткість може бути ключем до розуміння здатності людини справлятися з задачами, що занадто складні для вирішення на ЕОМ. Розвиток досліджень в області нечіткої математики призвів до появи нечіткої логіки і нечітких висновків, що виконуються з використанням знань, представлених нечіткими множинами, нечіткими відношеннями, нечіткими відповідностями.

3.2 Процес і система нечіткого логічного виведення

Загальне логічне виведення здійснюється в чотири етапи.

1. **Нечіткість** (введення нечіткості, фазифікація). Функції належності, визначені на входних змінних, застосовуються до їх фактичних значень для визначення міри істинності кожної передумови кожного правила.

2. **Логічне виведення.** Обчислене значення істинності для передумов кожного правила застосовується до висновків кожного правила. Це призводить до однієї нечіткої підмножини, яка буде призначена кожній змінній виведення для кожного правила.

У логічному виведенні МІНІМУМУ функція належності виведення «відсікається» по висоті, відповідній обчисленій мірі істинності передумови правила (нечітка логіка «ТА»). У логічному виведенні МНОЖЕННЯ функція

належності виводу масштабується за допомогою обчисленої міри істинності передумови правила.

3. Композиція. Всі нечіткі підмножини, призначені до кожної змінної виведення (у всіх правилах), об'єднуються разом, щоб сформувати одну нечітку підмножину для кожної змінної виведення. При подібному об'єднанні зазвичай використовуються операції МАКСИМУМ або СУМА.

При композиції МАКСИМУМУ комбіноване виведення нечіткої підмножини конструюється як поточковий максимум по всіх нечітких підмножинах (нечітка логіка «АБО»). При композиції СУМИ комбіноване виведення нечіткої підмножини конструюється як поточкова сума по всіх нечітких підмножинах, призначених змінній виведення правилами логічного виводу.

4. Приведення до чіткості (дефазифікація), використовується, коли корисно перетворити нечіткий набір висновків в чітке число. Існує велика кількість методів приведення до чіткості.

Система нечіткого виведення складається з п'яти функціональних блоків (рис. 77):

- **блок фазифікації**, що перетворює чисельні входні значення в ступінь відповідності лінгвістичним змінним;
- **база правил**, що містить набір нечітких правил типу «якщо-то»;
- **база даних**, у якій визначені функції належності нечітких множин, що використовуються в нечітких правилах;
- **блок прийняття рішень**, який виконує операції виведення на основі існуючих правил;
- **блок дефазифікації**, що перетворює результати виведення в чисельні значення.

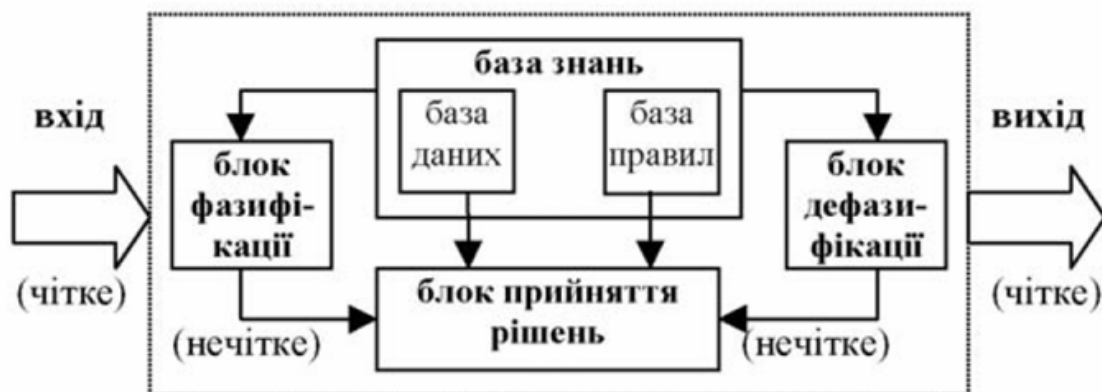


Рисунок 77 – Система нечіткого виведення

Виділяють три основних *типи систем нечіткого виведення*:

- **1-й тип**: вихідне значення знаходиться як зважене середнє результатів виконання кожного правила, для кожного з яких дефазифікація проводиться окремо, для таких систем вихідні функції належності повинні бути монотонно-неспадаючими;

- **2-й тип**: вихідне нечітке значення – це результат об'єднання нечітких виходів кожного правила, кожний нечіткий вихід зважено за допомогою ваг спрацьовування правил, чітке вихідне значення знаходиться в результаті дефазифікації об'єданого нечіткого виходу;

- **3-й тип**: система, побудована на правилах типа Сугено, вихідне значення є лінійною комбінацією вхідних значень плюс деяке постійне значення, загальний вихід є середнім зваженим всіх правил.

У загальному випадку як значення вхідних та вихідних змінних правил можна використовувати нечіткі множини, з якими не пов'язано ніяке поняття – оскільки при проведенні нечіткого виведення нечіткі терми все одно подаються нечіткими множинами і пов'язане з нечітким термом поняття не відіграє ніякої ролі.

Фазифікація – це визначення ступеня виконання антецедентів правил.

За допомогою фазифікації чіткому значенню ставляться у відповідність ступені його належності до нечітких множин.

Дефазифікація – це процедура перетворення нечіткої множини в чітке число за ступенем належності.

3.3 Методи нечіткого виведення

Розглянемо такі найбільш часто використовувані модифікації алгоритму нечіткого виведення, вважаючи, для простоти, що базу знань організовують два нечітких правила вигляду (3.2), (3.3):

$$П_1: \text{якщо } x \in A_1 \text{ і } y \in B_1, \text{ то } z \in C_1, \quad (3.2)$$

$$П_2: \text{якщо } x \in A_2 \text{ і } y \in B_2, \text{ то } z \in C_1, \quad (3.3)$$

де x, y – імена вхідних змінних;

z – ім'я змінної виведення;

A_1, B_1, C_1 – задані функції належності, при цьому чітке значення z_0 необхідно визначити на основі приведеної інформації та чітких значень x_0 і y_0 .

Алгоритм Мамдані. Даний алгоритм може бути описаний таким чином.

Нечіткість: знаходяться міри істинності для передумов кожного правила: $A_1(x_0), A_2(x_0), B_1(y_0), B_2(y_0)$.

Нечітке виведення: знаходяться рівні «відсікання» для передумов кожного з правил (з використанням операції МІНІМУМ (3.4)):

$$\begin{aligned}\alpha_1 &= A_1(x_0) \wedge B_1(y_0), \\ \alpha_2 &= A_2(x_0) \wedge B_2(y_0)\end{aligned}\tag{3.4}$$

потім знаходяться «усічені» функції належності (3.5):

$$\begin{aligned}C'_1(z) &= (a_1 \wedge C_1(z)), \\ C'_2(z) &= (a_2 \wedge C_2(z)).\end{aligned}\tag{3.5}$$

Композиція: з використанням операції МАКСИМУМ виконується об'єднання знайдених усічених функцій, що призводить до здобуття підсумкової нечіткої підмножини для змінної виведення з функцією належності (3.6):

$$\mu_{\Sigma}(z) = C(z) = C'_1(z) \vee C'_2(z) = (a_1 \wedge C_1(z)) \vee (a_2 \wedge C_2(z)).\tag{3.6}$$

Приведення до чіткості (знаходження z_0) проводиться, наприклад, методом центру тяжіння (центроїдним методом).

Алгоритм Сугено. Сугено і Такагі використовували набір правил в наступній формі (як і раніше, наводиться приклад з двох правил) (3.7), (3.8):

$$П_1: \text{якщо } x \in A_1 \text{ і } y \in B_1, \text{ то } z_1 = a_1x + b_1y;\tag{3.7}$$

$$П_2: \text{якщо } x \in A_2 \text{ і } y \in B_2, \text{ то } z_2 = a_2x + b_2y.\tag{3.8}$$

Даний алгоритм може бути описаний таким чином (рис. 78).

Перший етап – як в алгоритмі Мамдані.

На другому етапі знаходяться (3.9)

$$\begin{aligned}\alpha_1 &= A_1(x_0) \wedge B_1(y_0), \\ \alpha_2 &= A_2(x_0) \wedge B_2(y_0)\end{aligned}\quad (3.9)$$

та індивідуальні виходи правил (3.10)

$$z_1^* = a_1 x_0 + b_1 y_0, \quad z_2^* = a_2 x_0 + b_2 y_0. \quad (3.10)$$

На третьому етапі визначається чітке значення змінної виведення (3.11):

$$z_0 = \frac{\alpha_1 z_1^* + \alpha_2 z_2^*}{\alpha_1 + \alpha_2}. \quad (3.11)$$

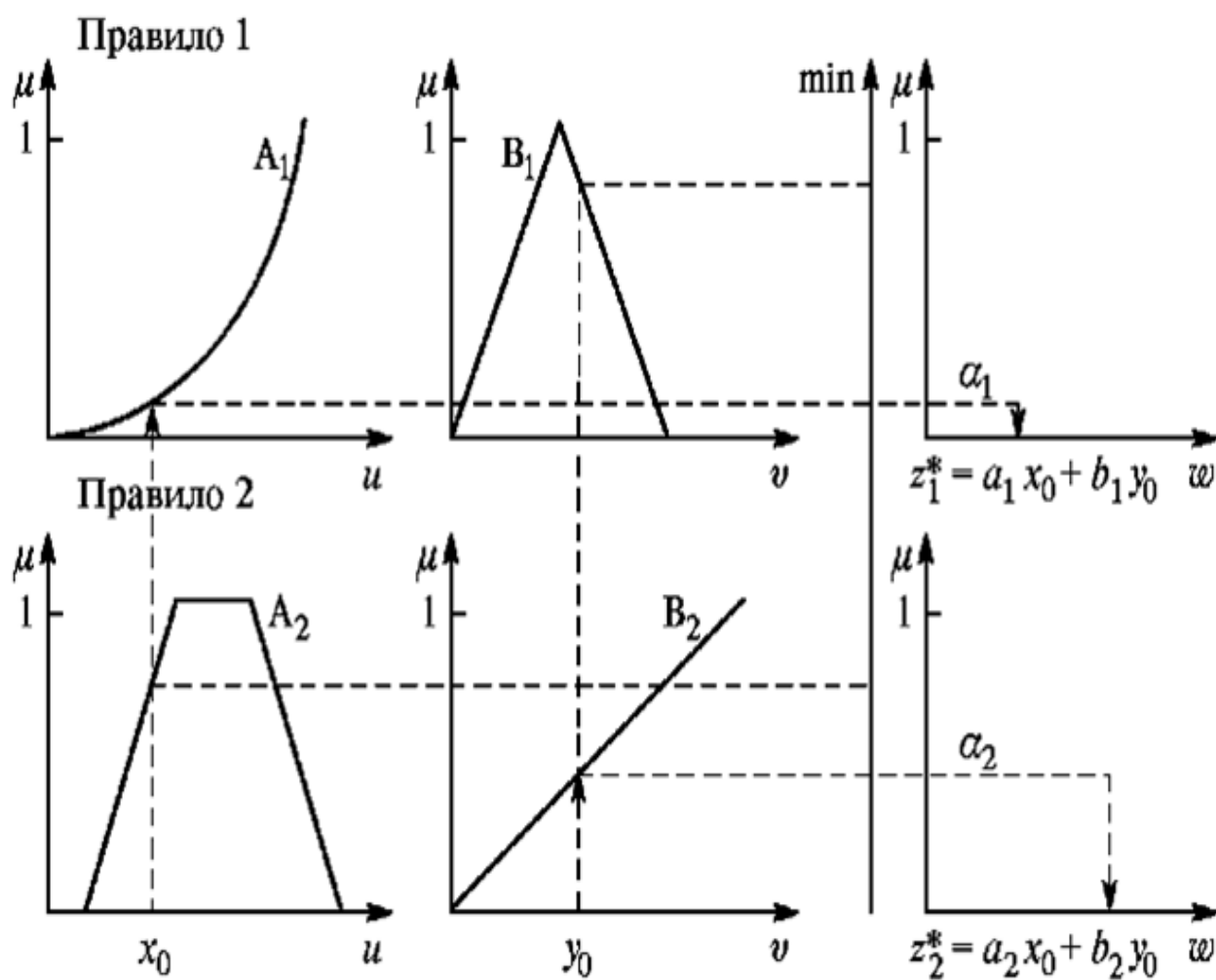


Рисунок 78 – Алгоритм Сугено

Алгоритм Ларсена. У алгоритмі Ларсена нечітка імплікація моделюється з використанням оператора множення.

Даний алгоритм може бути описаний таким чином (рис. 79).

Перший етап – як в алгоритмі Мамдані.

На другому етапі, як в алгоритмі Мамдані, знаходяться значення (3.12):

$$\begin{aligned}\alpha_1 &= A_1(x_0) \wedge B_1(y_0), \\ \alpha_2 &= A_2(x_0) \wedge B_2(y_0),\end{aligned}\tag{3.12}$$

а потім часткові нечіткі підмножини (3.13):

$$\alpha_1 C_1(z), \quad \alpha_2 C_2(z).\tag{3.13}$$

Знаходиться підсумкова нечітка підмножина з функцією належності (3.14):

$$\mu_{\Sigma}(z) = C(z) = (a_1 C_1(z)) \vee (a_2 C_2(z)).\tag{3.14}$$

Приведення до чіткості (як в раніше розглянутих алгоритмах).

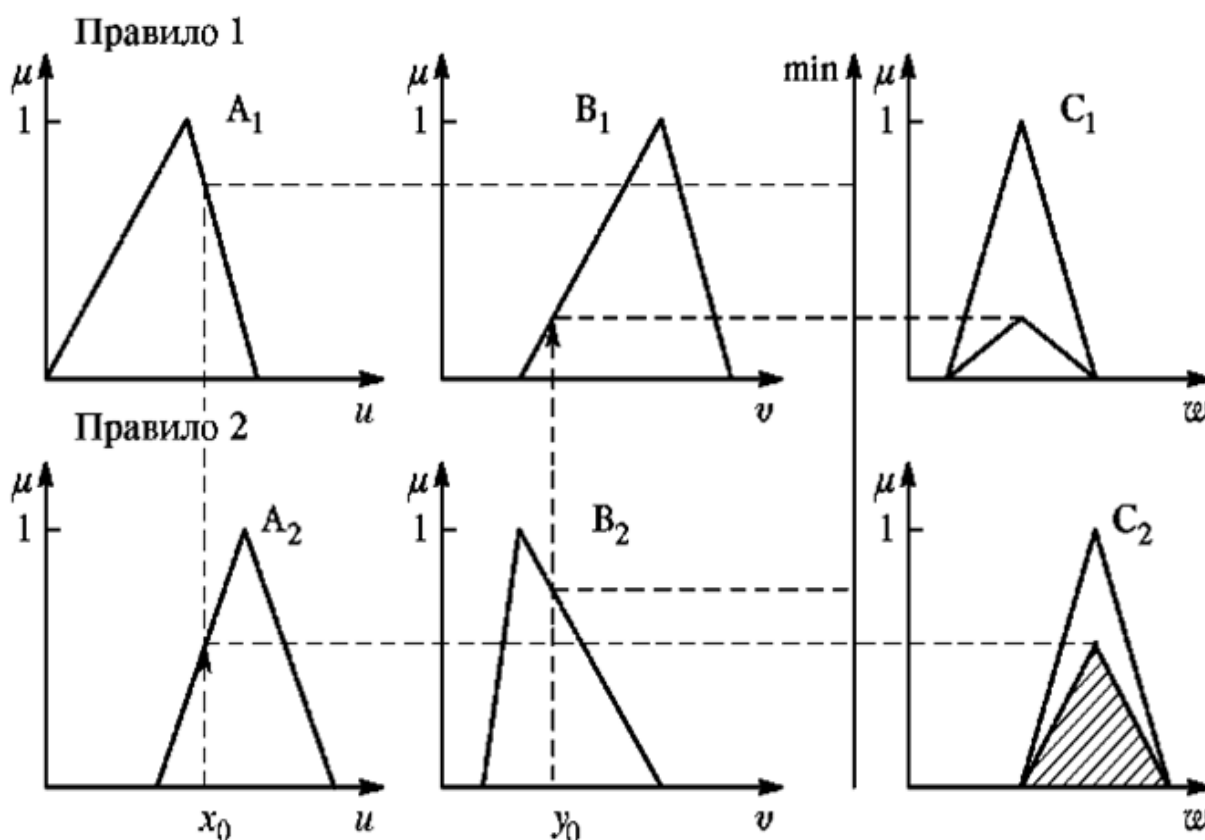


Рисунок 79 – Алгоритм Ларсена

3.4 Метод резолюцій

Правило резолюції в логіці висловлювань має таку структуру (3.15):

$$\frac{A \vee B, \neg A \vee C}{B \vee C}, \quad (3.15)$$

де A , B і C – довільні формули логіки висловлювань.

Висловлювання під рискою є наслідком висловлювань над рискою.

Неважко перевірити, що істинність нижнього судження впливає з істинності верхніх посилок.

Алгоритм методу резолюцій. Задано множину гіпотез f_1, f_2, \dots, f_n і висновок g . Алгоритм дає змогу визначити, чи є формула g логічним наслідком множини гіпотез f_1, f_2, \dots, f_n .

Крок 1. Побудувати кон'юнкцію множини гіпотез f_1, f_2, \dots, f_n і заперечення висновку g_u у вигляді $f_1 \wedge f_2 \wedge \dots \wedge f_n \wedge g_u$. Звести отриману формулу до кон'юнктивної нормальної форми і записати множину її елементарних диз'юнкцій S .

Крок 2. Записати кожен елементарну диз'юнкцію множини S в окремому рядку.

Крок 3. Вибрати дві елементарні диз'юнкції, які містять пару літералів, і побудувати їх резольвенту. Записати одержану резольвенту в новому рядку, якщо в попередніх рядках ще немає такої елементарної диз'юнкції.

Крок 4. Крок 3 виконувати до отримання диз'юнкції з рангом 0. Одержання елементарної диз'юнкції з рангом 0 свідчить про те, що формулу g можна вивести з f_1, f_2, \dots, f_n . Якщо неможливо отримати резольвенту, відмінну від елементів множини S і вже побудованих резольвент, то множина S неспростовна. Кінець.

3.5 Експертні системи

Експертною системою (ЕС) називають систему підтримки прийняття рішень, яка містить знання з певної вузької предметної області, а також може пропонувати користувачу рішення проблем з цієї галузі і обґрунтовувати їх. Експертна система складається з бази знань, механізму логічного виводу і підсистеми обґрунтувань.

Експертна система акумулює професійні знання керівників і фахівців, використовуючи їх для формування бази знань, яка містить набір взаємопов'язаних правил. При прийнятті рішень стає можливим аналіз наслідків різних рішень у вигляді питань «що буде, якщо...», не витрачаючи часу на трудомісткий процес програмування.

Створення експертних систем – це спроба значного розширення області застосування комп'ютерної техніки і суттєвого збільшення її можливостей як допомоги людині у її інтелектуальній роботі.

До появи експертних систем комп'ютери створювались за принципами алгоритмічної методології. Для того, щоб такі обчислювальні системи могли успішно працювати, вимагається виконати значну низку попередніх умов. Перш за все, для кожної розв'язуваної задачі потрібно знайти або створити алгоритм. Потім цей алгоритм потрібно перетворити у докладну програму яка реалізуватиме майбутні обчислення. Також, потрібно потурбуватися про те, щоб всі обчислення були забезпечені повним обсягом достовірної вихідної інформації.

Такий спосіб використання комп'ютерної техніки пов'язаний зі значними труднощами. По-перше, по мірі зростання складності розв'язуваних задач, швидко зростає трудомісткість і вартість програмування роботи комп'ютера, що вже зараз стає сильним гальмом для подальшого використання обчислювальної техніки. По-друге, багато практичних задач, які виникають в процесі діяльності людини, не забезпечені належним об'ємом вихідних даних, оскільки людина діє як правило в умовах більшої чи меншої інформаційної невизначеності.

Як результат, багато важливих задач людина не може перекласти на комп'ютер, що працює за принципами алгоритмічної методології, що істотно знижує область практичного застосування комп'ютерної техніки. Разом з цим, людина успішно справляється з подібними задачами завдяки своєму вмінню:

- працювати з задачами, не здійснюючи їх повну формалізацію;
- знаходити і використовувати для розв'язання задач різноманітні, найнеочікуваніші джерела інформації;
- поєднувати суперечливі відомості, надаючи їм потрібну інтерпретацію і відповідну вагу;

- цілеспрямованого вивчення або перевірки об'єкта дії для одержання додаткової інформації;

- відтермінувати рішення до накопичення потрібних даних;

- продукувати неоднозначні рішення і успішно керуватися ними;

- створювати і накопичувати знання, які уможлиблюють діяти в умовах значної інформаційної невизначеності.

Спроба наділити комп'ютери переліченими цінними характеристиками призвела до створення експертних систем.

На практиці експертна система представляє собою спеціалізовану обчислювальну машину (процесор), що відтворює алгоритм розв'язання людиною певних практичних задач на основі професійно-орієнтованих знань, переданих їй відповідними спеціалістами.

При цьому експертна система проявляє такі властивості:

- по мірі розв'язання задач проводить діалог з людиною, обмінюючись з ним питаннями і відповідями;

- аналізує наявну проблемну ситуацію і може управляти нею через людину;

- обґрунтовує зроблені висновки і пропоновані дії у зрозумілій для людини формі;

- сприймає і накопичує нові професійні знання.

Особливістю сучасних діючих експериментальних експертних систем є їх дуже вузька спеціалізація. Це системи для діагностики певного конкретного виду захворювань людини, або для визначення структурної формули певного класу органічних з'єднань, або для пошуку оптимальної конфігурації конкретної обчислювальної системи.

Вузька спеціалізація експериментальних експертних систем викликана бажанням зменшити об'єм професійних знань, що закладаються в систему, для спрощення задачі створення цих знань та їх збереження у пам'яті ІС.

Професійні знання передаються експертній системі відповідним спеціалістом, а їх зведення до вигляду, зручного для використання у комп'ютері виконує програміст. Найпоширенішою і природною формою представлення знань у системі є їх запис у вигляді професійних правил або тверджень типу «якщо..., то...». Ліва частина такого правила представляє поєднання фактів або ознак, які характеризують деяку умову, а права частина вказує на дію або висновок, що відповідає за досвідом спеціаліста наявній ситуації.

Знання або, за термінологією спеціалістів, база знань експертної системи складається з великої кількості подібних професійних правил різного ступеня спільності. Розв'язуючи задачу, експертна система вибирає правила у порядку зниження їх спільності, що відтворює алгоритм міркувань спеціаліста у подібній ситуації від цілі до конкретних дій.

Діалог з системою та її поради будуть зовсім непростими, якщо вони стосуватимуться малознайомої для людини професійної області, а сама порада буде результатом вибірки зі значної множини альтернатив. Цей непростий вибір утворюється шляхом багатократного вводу у систему інформації за її запитам.

Експертна система, побудована таким чином, має низку певних переваг. По-перше, програмування системи здійснюється на більш зрозумілому для людини рівні, ніж у сучасних комп'ютерах.

Це робить експертні системи та їх програмування доступним для малопідготовлених користувачів. По-друге, на відміну від сучасних комп'ютерів, експертна система може пояснити людині, яким чином вона отримала той чи інший результат. По-третє, експертна система, база знань якої побудована на основі знань групи спеціалістів, має більші інтелектуальні можливості, ніж кожний спеціаліст окремо. По-четверте, експертна система просто навчається шляхом поповнення її бази знань новими знаннями, що набуваються спеціалістами. Надалі, систему можна наділити здатністю до самонавчання.

Для представлення знань у експертних системах використовують найрізноманітніші способи. Найпоширеніші способи представлення знань використовують продукції (семантичні мережі). Продукція представляє порцію (квант) знання у формі правила типу «якщо..., то...».

Семантичні мережі побудовані з понять і створених між ними зв'язків, прикладом яких можуть бути зв'язки між поняттями «людина», «особа», «чоловік», «жінка». З різних видів семантичних мереж найпоширенішими є мережі з фреймів, які є моделями понять природної мови.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Методи, моделі та інформаційні технології оцінювання станів складних об'єктів: монографія / Є. І. Кучеренко, В. Є. Кучеренко, І. С. Глушенкова, І. С. Творошенко; ХНАМГ, ХНУРЕ. – Х. : ХНАМГ; ХНУРЕ, 2012. – 278 с.
2. Люгер Дж. Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Дж. Ф. Люгер. – М. : Издательский дом «Вильямс», 2003. – 864 с.
3. Бодянський Є. В. Нейро-фаззі моделі в системах штучного інтелекту: навч. посібник / Є. В. Бодянський, Є. І. Кучеренко. – Х. : ХНУРЕ, 2006. – 177 с.
4. Кучеренко Є. І. Сіткові моделі в задачах аналізу складних систем: навч. посібник для вузів / Є. І. Кучеренко. – Х. : ХТУРЕ, 1999. – 99 с.
5. Гаврилова Т. А. Базы знаний интеллектуальных систем / Т. А. Гаврилова, В. Ф. Хорошевский. – СПб. : Питер, 2000. – 384 с.
6. Томашевський В. М. Моделювання систем / В. М. Томашевський. – К. : Видавнича група ВНУ, 2005. – 352 с.
7. Кучеренко Е. И. Прикладные аспекты моделирования нечетких процессов в сложных системах / Е. И. Кучеренко, И. С. Творошенко // Сборник научных трудов ХУВС. – 2010. – Вып. 1 (23). – С. 127-131.
8. Советов Б. Я. Моделирование систем / Б. Я. Советов, С. А. Яковлев. – М. : Высш. Школа, 2001. – 343 с.
9. Методи, моделі та інформаційні технології оцінювання станів складних об'єктів: монографія [Електронний ресурс]. – Режим доступу: <http://eprints.kname.edu.ua/29457/>.
10. IDEF Family of Methods. A Structured Approach to Enterprise Modeling & Analysis [Електронний ресурс]. – Режим доступу: www.idef.com.
11. Цифровий репозиторій ХНУМГ ім. О. М. Бекетова [Електронний ресурс]. – Режим доступу: <http://eprints.kname.edu.ua>.
12. Державний стандарт України ДСТУ 3008-95 «Документація. Звіти у сфері науки і техніки. Структура і правила оформлення».
13. ДСТУ ГОСТ 7.1: 2006 «Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання».

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних, розрахунково-графічних
та самостійних робіт з дисципліни

ОСНОВИ МОДЕЛЮВАННЯ СКЛАДНИХ СИСТЕМ

*(для студентів 2 курсу денної та заочної форм навчання
напрямку підготовки 6.080101 – Геодезія, картографія та землеустрій)*

Укладач **ТВОРОШЕНКО** Ірина Сергіївна

Відповідальний за випуск *К. А. Мамонов*

За авторською редакцією

Комп'ютерний набір *І. С. Творошенко*

Комп'ютерне верстання *І. В. Волосожарова*

План 2014, поз. 55 М

Підп. до друку 15.04.2015 р.

Друк на різнографі.

Зам. №

Формат 60x84/16

Ум. друк. арк. 5,8

Тираж 50 пр.

Видавець і виготовлювач:

Харківський національний університет
міського господарства імені О. М. Бекетова,

вул. Революції, 12, Харків, 61002

Електронна адреса: rectorat@kname.edu.ua

Свідоцтво суб'єкта видавничої справи:

ДК № 4705 від 28.03.2014 р.